

Auditoría compatibilidad producto software

Matías Sánchez-Carrasco García

GRADO EN INGENIERÍA INFORMÁTICA

FACULTAD DE INFORMÁTICA

DEPARTAMENTO DE ARQUITECTURA DE COMPUTADORES Y AUTOMÁTICA

UNIVERSIDAD COMPLUTENSE DE MADRID



TRABAJO FIN DE GRADO EN TECNOLOGÍA DE LA INFORMACIÓN

2014/2015

Directores: Pedro Luis García Repetto y M. Carmen Molina Prego



A Pedro, por confiar en mí hasta el final.

Autorización de difusión y utilización

Los abajo firmantes, alumno y tutores del Trabajo Fin de Grado (TFG) en el Grado en Ingeniería Informática de la Facultad de Informática, autorizan a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el Trabajo Fin de Grado (TFG) cuyos datos se detallan a continuación. Así mismo autorizan a la Universidad Complutense de Madrid a que sea depositado en acceso abierto en el repositorio institucional con el objeto de incrementar la difusión, uso e impacto del TFG en Internet y garantizar su preservación y acceso a largo plazo.

TÍTULO del TFG: Auditoría compatibilidad producto software

Curso académico: 2014/ 2015

Nombre del Alumno:

Matías Sánchez-Carrasco García

Tutores del TFG y departamentos a los que pertenecen:

Pedro Luis García Repetto, departamento de arquitectura de computadores y automática

M. Carmen Molina Prego, departamento de arquitectura de computadores y automática

Firma del alumno

Firmas de los tutores

Matías Sánchez-Carrasco García

Pedro Luis García Repetto

M. Carmen Molina Prego

Madrid, 14 de Septiembre del 2015

Índice

Autorización de difusión y utilización	I
Índice.....	III
Índice de figuras:.....	VII
Resumen.....	IX
Abstract	X
1. Introducción	1
1.1 Compatibilidad.....	1
1.2 Proceso de auditoría según ISO/IEC 19011	1
1.3 Objetivos y finalidad del TFG.....	1
1.3.1 Análisis del concepto de compatibilidad del producto software	1
1.3.2 Definir un referente de auditoría sobre la compatibilidad del producto software	1
1.3.3 Definir un proceso de auditoría sobre la compatibilidad del producto software.....	2
1.4 Metodología	2
1.4.1 Análisis detallado de antecedentes	2
1.4.1.1 Artículos de investigación y tesis	2
1.4.1.2 IEEE.....	2
1.4.1.3 ISO/IEC.....	3
1.4.1.3.1 ISO/IEC 9126.....	3
1.4.1.3.2 ISO/IEC 14598.....	3
1.4.1.3.3 Familia ISO/IEC 25000.....	4
1.5 Estructura del trabajo	4
2. Fundamentos	5
2.1 ISO/IEC 9126.....	5
2.1.1 Apartados de la ISO/IEC 9126	5
2.1.1.1 ISO 9126-1 Modelo de calidad.	5
2.1.1.2 ISO 9126-2 Métricas externas.	6
2.1.1.3 ISO 9126-3 Métricas internas.	6
2.1.1.4 ISO 9126-4 Métricas de calidad en el uso	7
2.1.2 Características definidas por la ISO/IEC 9126.....	7
2.1.2.1 Funcionalidad.....	7
2.1.2.2 Fiabilidad	8
2.1.2.3 Usabilidad	8

2.1.2.4 Eficiencia	8
2.1.2.5 Mantenibilidad	8
2.1.2.6 Portabilidad	9
2.2 ISO/IEC 14598	9
2.2.1 ISO/IEC 14598-4	10
2.2.1.1 ISO/IEC 14598-4: La portabilidad como métrica externa	11
2.2.1.2 ISO/IEC 14598-4: La portabilidad como métrica en la calidad de uso	12
2.2.1.3 ISO/IEC 14598-4: La portabilidad como método de evaluación	13
2.2.2 ISO/IEC 14598-5	13
2.3 ISO/IEC 25000	14
2.3.1 ISO/IEC 25010	15
2.3.1 ISO/IEC 25040	16
3. Definición de un referente de auditoría	19
3.1 Introducción	19
3.2 Métricas externas de portabilidad definidas en la ISO 9126-2	19
3.2.1 Métricas externas de adaptabilidad	19
3.2.1.1 Adaptabilidad de estructuras de datos	20
3.2.1.2 Adaptabilidad al entorno Hardware	20
3.2.1.3 Adaptabilidad al entorno organizacional	20
3.2.1.4 Amigabilidad con el usuario para la portabilidad	20
3.2.1.5 Adaptabilidad al entorno del sistema software	21
3.2.2 Métricas externas de instalabilidad	21
3.2.2.1 Facilidad de Instalación	21
3.2.2.2 Facilidad para reintentar la configuración	21
3.2.3 Métricas externas de coexistencia	22
3.2.3.1 Posibilidad de coexistencia	22
3.2.4 Métricas externas de reemplazabilidad	22
3.2.4.1 Uso continuado de los datos	22
3.2.4.2 Inclusividad de funciones	23
3.2.4.3 Consistencia funcional al soporte del usuario	23
3.2.5 Métricas externas de cumplimiento de la portabilidad	23
3.2.5.1 Cumplimiento de la portabilidad	23
3.3 Métricas internas de portabilidad definidas en la ISO 9126-3	24

3.3.1 Métricas internas de adaptabilidad	24
3.3.1.1 Adaptabilidad de las estructuras de datos	24
3.3.1.2 Adaptabilidad al entorno de hardware	24
3.3.1.3 Adaptabilidad al entorno organizacional	25
3.3.1.4 Amigabilidad con el usuario para la portabilidad	25
3.3.1.5 Adaptabilidad al entorno del sistema software	25
3.3.2 Métricas internas de instalabilidad	25
3.3.2.1 Facilidad para reintentar la configuración	26
3.3.2.2 Esfuerzo en la instalación	26
3.3.2.3 Flexibilidad en la instalación	26
3.3.3 Métricas internas de coexistencia	26
3.3.3.1 Posibilidad de coexistencia	26
3.3.4 Métricas internas de reemplazabilidad,	27
3.3.4.1 Uso continuado de los datos	27
3.3.4.2 Inclusividad de funciones	27
3.3.5 Métricas internas de cumplimiento de la portabilidad	27
3.3.5.1 Cumplimiento de la portabilidad	27
3.4 Nuevas métricas de portabilidad	28
3.4.1 Adaptabilidad a distintos dispositivos	28
3.4.2 Adaptabilidad al servidor de aplicaciones	28
3.4.3 Instalabilidad frente al reemplazo del sistema operativo	29
3.4.4 Adaptabilidad ante el cambio del <i>framework</i> de desarrollo	30
3.4.5 Métricas para la adaptabilidad del código	30
3.4.5.1 Autocontención o encapsulamiento de módulos	30
3.4.5.2 Autodocumentación de formas estructurales	30
3.4.5.3 Especificación de módulos	31
3.5 Tabla resumen de las métricas	31
4. Caso práctico	33
4.1. Introducción	33
4.2 Descripción del laboratorio de auditoría y aplicaciones a ser auditadas	33
4.2.1 Laboratorio de auditoría	33
4.2.2 Aplicaciones a ser auditadas	34
4.3 Objetivos	34

4.4 Método (actividades del 25040).....	34
4.4.1 Establecer los requisitos de la evaluación	35
4.4.2 Especificar la evaluación	35
4.4.3 Diseñar la evaluación.....	36
4.4.4 Ejecutar la evaluación.....	37
4.4.5 Concluir la evaluación:	38
4.5 Conclusiones	39
5. Conclusiones y líneas abiertas	41
5.1 Conclusiones	41
5.2 Líneas abiertas.....	41
Anexo A: Tabla resumen de las métricas	43
Anexo B: Plantilla para <i>checklist</i> de adaptabilidad.....	49
Anexo C: <i>Checklist</i> UCM	51
Anexo D: <i>Checklist</i> UAM.....	53
Anexo E: <i>Checklist</i> UC3M	55
Anexo F: <i>Checklist</i> UOC	57
Bibliografía	59

Índice de figuras:

Ciclo de vida del software según la ISO 12207	3
Relación entre la calidad externa, interna y calidad de uso.	6
Tabla B.2: Especificación de las características, subcaracterísticas y métricas externas de la calidad del software, con ejemplos.....	11
Tabla B.3: Especificación de las métricas de calidad de uso, con ejemplos	12
Tabla B.4: Ranking coste-eficiencia de métodos de evaluación.....	13
Encuesta de la AIMC que muestra en percentiles los equipos de acceso a Internet y su evolución ...	33
Gráficos de resultados.....	38

Resumen

Este Trabajo de Fin de Grado (TFG en adelante) se ha centrado en la compatibilidad del producto software. La compatibilidad o portabilidad es una de las características de la calidad software contemplada en los estándares o referentes de calidad software. Se puede definir la compatibilidad como la capacidad de un producto software para ser transferido desde un entorno a otro.

La primera aportación de este trabajo de fin de grado es presentar los resultados exhaustivos y detallados fruto del proceso de investigación y análisis de los estándares siguientes: ISO/IEC 9126, ISO/IEC 14598 y la familia ISO/IEC 25000. Estos estándares no incluyen ejemplos prácticos. Debido a ello en este TFG se aportan para las métricas analizadas ejemplos reales que faciliten su comprensión.

En el análisis llevado a cabo se han detectado debilidades en dichos estándares; en concreto áreas de la portabilidad no tratadas por los mismos. Por ello en este TFG se propone como segunda aportación nuevas métricas que persiguen mejorar los resultados a obtener en las auditorías de calidad del producto software. De esta forma se confecciona un nuevo referente de auditoría de producto software al aunar las métricas existentes en los estándares mencionados con las nuevas métricas aportadas en el proceso de investigación realizado.

En primer lugar en este documento se hace una introducción que da una idea general del trabajo, después se analizan con detalle los fundamentos de este trabajo: ISO/IEC 9126, ISO/IEC 14598 y la familia ISO/IEC 25000 y su relación con la portabilidad. Más adelante se analizan las métricas que se utilizan en el referente ISO/IEC 9126 y se proponen nuevas métricas para la realización de una auditoría de calidad.

Como tercera aportación y para poner en práctica algunas de estas métricas se propone un proceso de auditoría especializado en la compatibilidad o portabilidad del producto software. Este proceso se traslada a un caso práctico en el que se analiza la portabilidad de cuatro aplicaciones web en distintos dispositivos, sistemas operativos y navegadores. Por último se muestran una serie de conclusiones y se proponen una serie de líneas abiertas para el estudio de la compatibilidad del producto software.

Palabras clave: Portabilidad, compatibilidad, transferibilidad, auditoría informática, ISO/IEC 9126, ISO/IEC 14598, ISO/IEC 25000, adaptabilidad.

Abstract

This final degree dissertation work (FDDW hereinafter) has focused on the compatibility of software product. The compatibility or portability is one of the features of the software quality standards relating to the software quality. Compatibility can be defined as the ability of a software product to be transferred from one environment to another.

The first contribution of this work presents the detailed comprehensive results of the process of investigation and analysis of the following standards: ISO/IEC 9126, ISO/IEC 14598 and the ISO/IEC 25000 family. These not include practical examples. As a result in this FDDW provides metrics analysis and real examples that facilitate understanding.

In the analysis carried out along of the work, we have been detected certain part of weakness in those standards; specifically areas like portability they don't treat by the same. Reason because this work proposes as second contribution new metrics that pursue the goal to improve the results that can obtain the audit of software product quality. Thereby a new standard of auditing software product join the existing metrics standard with new metrics provided in the process of this research.

First of all in this document, an introduction that gives a general view of that work has done, and then analysed in details the basics of this work: The ISO/IEC 9126, the ISO/IEC 14598 and the ISO/IEC 25000 family and its relationship with the portability. Further on, the metrics used in the ISO/IEC 9126, are being analysed and proposed new metrics for the analyzation of a high quality audit.

As a third contribution and to give effect of some of those metrics, a specialized auditing is being proposed in compatibility and portability. These process moves to a practical case in which four web applications portability across devices, operating systems and browser are analysed. Finally some conclusions are shown and other one is being proposed some guidelines for the study of the compatibility of the software product.

Keywords: Portability, compatibility, transferability, information technology audit, ISO/IEC 9126, ISO/IEC 14598, ISO/IEC 25000, adaptability.

1. Introducción

1.1 Compatibilidad

Según el diccionario de la Real Academia, la compatibilidad es la “Cualidad de compatible”, y compatible se define en su segunda acepción como “Dicho de una persona o de una cosa: Que puede estar, funcionar o coexistir sin impedimento con otra”. Esta definición también es aplicable al producto software. La ISO/IEC 9126 define la compatibilidad como la capacidad de un producto software para ser transferido desde un entorno a otro.

La compatibilidad es un factor importante a tener en cuenta a la hora de diseñar un producto software, ya que ahorra costes a la hora de transferir el producto, además de llegar a un mayor número de clientes potenciales.

1.2 Proceso de auditoría según ISO/IEC 19011

La ISO/IEC 19011 proporciona unas directrices para la auditoría de sistemas de gestión y define un proceso sistemático, independiente y documentado para obtener evidencias de la auditoría y evaluarlas de manera objetiva con el fin de determinar la extensión en la que se cumplen los criterios de auditoría.

Este referente establece una serie de actividades para la preparación y realización de la auditoría como parte de un programa de auditoría. Son las siguientes:

1. Inicio de la auditoría
2. Preparación de las actividades de auditoría
3. Realización de las actividades de auditoría
4. Preparación y distribución del informe de auditoría
5. Finalización de la auditoría
6. Realización de actividades de seguimiento

También indica cómo hacer un programa de auditoría y qué competencias debe tener un auditor y cómo evaluar dichas competencias.

1.3 Objetivos y finalidad del TFG

1.3.1 Análisis del concepto de compatibilidad del producto software

El primer objetivo de este trabajo es realizar un análisis profundo del concepto de compatibilidad del producto software. Este concepto puede llegar a ser bastante complejo debido a los distintos referentes que la definen (9126 y 25010) y a las múltiples características que se le asignan.

El referente ISO/IEC 9126 se refiere a la compatibilidad como “Portabilidad”, mientras que el referente ISO/IEC 25010 desdobra el concepto de portabilidad en dos: transferibilidad y portabilidad.

1.3.2 Definir un referente de auditoría sobre la compatibilidad del producto software

Otro objetivo de este trabajo es proponer un nuevo referente de auditoría de compatibilidad o portabilidad del producto software. Este referente está formado por las métricas existentes en el

estándar ISO/IEC 9126, el cual es analizado con detenimiento en el segundo apartado de este trabajo, y por las nuevas métricas que se propondrán en este trabajo relacionadas con la portabilidad del producto software. En concreto métricas relacionadas principalmente con la adaptabilidad y con la instalabilidad en mayor medida.

1.3.3 Definir un proceso de auditoría sobre la compatibilidad del producto software

Se intentará mejorar el proceso de auditoría recogido en los estándares analizados. Este proceso de auditoría será completado con las nuevas métricas propuestas para así conseguir definir un proceso de auditoría actualizado y acorde a los nuevos dispositivos e innovaciones que van surgiendo a lo largo de los años con el proceso de la tecnología.

1.4 Metodología

1.4.1 Análisis detallado de antecedentes

1.4.1.1 Artículos de investigación y tesis

Para la realización de este trabajo se han consultado múltiples artículos de investigación y tesis, principalmente mediante buscadores de internet. Si bien han ayudado a comprender los referentes, se ha encontrado varios artículos que traten acerca de la portabilidad del producto software. No obstante, todos ellos la mencionaban, pero ninguno aportaba ningún análisis profundo o novedad a la investigación.

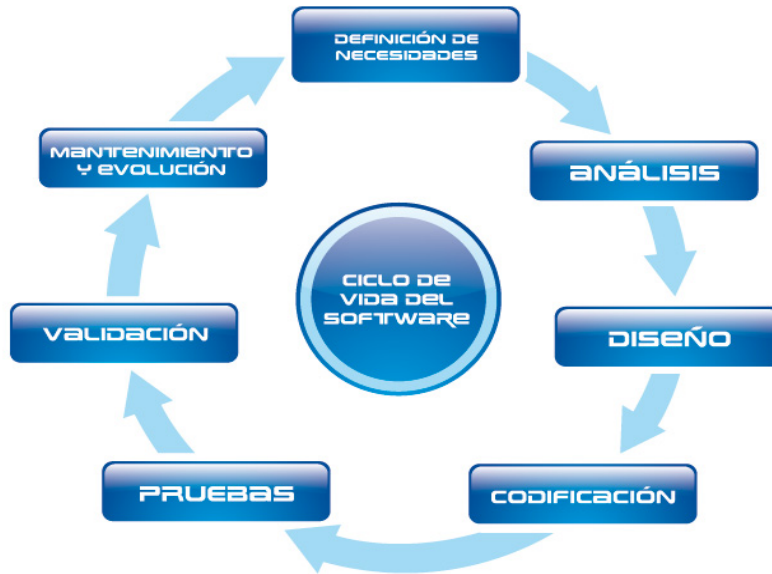
1.4.1.2 IEEE

El Instituto de Ingeniería Eléctrica y Electrónica (IEEE¹), es una asociación mundial de técnicos e ingenieros dedicada a la estandarización y el desarrollo en áreas técnicas, es la mayor asociación internacional sin ánimo de lucro formada por profesionales de las nuevas tecnologías.

La principal aportación del IEEE en este tema es la norma ISO/IEC 12207, la cual fue creada con el propósito de establecer un marco común para el ciclo de vida del software para:

- Adquirir, suministrar, desarrollar, operar y mantener software.
- Gestionar, controlar y mejorar el marco de trabajo del software.
- Como base para el comercio internacional de software.

¹ Comúnmente leído i e cubo



Ciclo de vida del software según la ISO 12207

1.4.1.3 ISO/IEC

1.4.1.3.1 ISO/IEC 9126

El referente ISO/IEC 9126 es una guía para la evaluación de la calidad del software, permite especificar y evaluar la calidad del software desde diferentes criterios asociados con adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad y auditoría de software. Los modelos de calidad para el software se describen mediante la calidad interna, externa y en el uso.

Este referente asigna una serie de características y subcaracterísticas de calidad interna, externa y en el uso y las define. Se podría decir que este referente tan solo define un modelo de calidad para el producto software.

1.4.1.3.2 ISO/IEC 14598

El referente ISO/IEC 14598 establece un marco de trabajo para evaluar la calidad de los productos de software proporcionando además, métricas y requisitos para los procesos de evaluación de los mismos.

En particular, es utilizada para aplicar los conceptos descritos en la norma ISO/IEC 9126. Se definen y describen las actividades necesarias para analizar los requisitos de evaluación, para especificar, diseñar y realizar acciones de evaluación y para concluir la evaluación de cualquier tipo de producto de software.

Este referente define las principales características del proceso de evaluación (Repetitividad, reproducibilidad, imparcialidad y objetividad). Para estas características se describe un proceso compuesto por una serie de actividades:

- Análisis de los requisitos de evaluación.

- Evaluación de las especificaciones.
- Evaluación del diseño y definición del plan de evaluación.
- Ejecución del plan de evaluación.
- Evaluación de la conclusión.

1.4.1.3.3 Familia ISO/IEC 25000

La familia de estándares ISO/IEC 25000, también conocida como “**Software Product Quality Requirements and Evaluation**” o más comúnmente, por sus siglas SQuaRE.

Esta familia reemplaza a la ISO/IEC 9126 y a la ISO/IEC 14598. Su objetivo es organizar, enriquecer y unificar las series que cubren dos procesos principales: Especificación de requisitos de calidad del software y evaluación de la calidad del software, soportada por el proceso de medición de calidad del software.

Aún se encuentra en desarrollo, no habiéndose publicado aún todos los referentes que la forman.

1.5 Estructura del trabajo

Este trabajo está estructurado en cinco capítulos, el primero de ellos es esta introducción la cual nos da una idea general del trabajo.

En el segundo capítulo se analizan con detalle los fundamentos y antecedentes de este trabajo: La ISO/IEC 9126, la ISO/IEC 14598 y la familia ISO/IEC 25000 y su relación con la portabilidad.

El tercer capítulo se analizan las métricas que se utilizan en el referente ISO/IEC 9126 y se proponen nuevas métricas para la realización de una auditoría de compatibilidad o portabilidad.

El cuarto capítulo es un caso práctico que analiza la portabilidad de cuatro aplicaciones web en distintos dispositivos, sistemas operativos y navegadores.

Por último, el quinto capítulo de este trabajo muestra una serie de conclusiones derivadas de los capítulos anteriores y propone una serie de líneas abiertas para el estudio y auditoría de la compatibilidad del producto software.

2. Fundamentos

En este apartado se realizará un análisis detallado de los antecedentes en las distintos referentes ISO/IEC que hablan sobre la compatibilidad. Son tres:

El referente **ISO/IEC 9126** tan solo menciona la compatibilidad como una característica de calidad del software y le atribuye un conjunto de subcaracterísticas sin ahondar más allá de definiciones. También incluye una lista de métricas internas y externas para cuantificar la portabilidad.

El referente **ISO/IEC 14598** se utiliza para aplicar los conceptos descritos en la ISO/IEC 9126, es aquí donde se da una guía para evaluar la calidad del software, también en el aspecto de la portabilidad

Y por último la familia **ISO/IEC 25000**, la cual está destinada a reemplazar los dos referentes anteriormente descritos, juntándolos y ampliándolos.

2.1 ISO/IEC 9126

El referente ISO/IEC 9126 Es un estándar internacional para la evaluación de la calidad del software. Proviene del modelo establecido en 1977 por McCall y sus colegas, se originó motivado por US Air Force y DoD (Departamento de defensa de Estados Unidos). Se focaliza en el producto final, identificando atributos claves desde el punto de vista del usuario estos atributos se denominan factores de calidad y son normalmente atributos externos.

El modelo de calidad McCall está organizado sobre tres tipos de Características de Calidad:

- Factores (especificar): Describen la visión externa del software, como es visto por los usuarios.
- Criterios (construir): Describen la visión interna del software, como es visto por el desarrollador.
- Métricas (controlar): Se definen y se usan para proveer una escala y método para la medida.

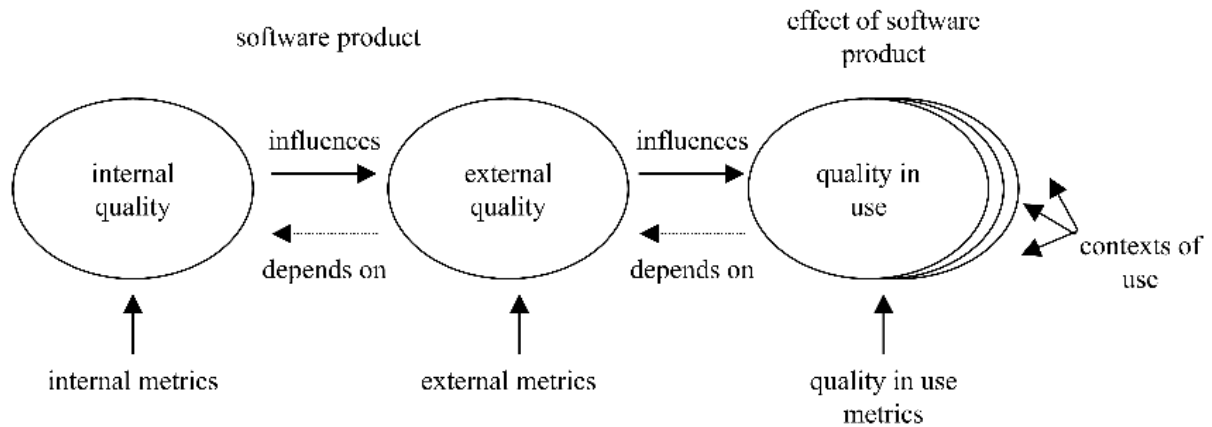
El referente ISO / IEC 9126 fue originalmente desarrollado en 1991 para proporcionar un esquema para la evaluación de calidad del software. La versión más reciente es del año 2004. Este referente está siendo reemplazado por el proyecto SQuaRE, ISO 25000:2005, el cual sigue los mismos conceptos.

2.1.1 Apartados de la ISO/IEC 9126

El referente ISO/IEC 9126 está dividido en cuatro partes: Modelo de calidad, métricas internas, métricas externas y métricas de calidad en el uso. En los siguientes subapartados se describen y analizan cada una de ellas.

2.1.1.1 ISO 9126-1 Modelo de calidad.

La primera parte del referente se centra en la definición de un modelo de calidad para cualquier tipo de producto software. Este modelo tiene dos partes: la calidad interna y externa y la calidad en uso.



Relación entre la calidad externa, interna y calidad de uso.

En este apartado del referente no se elabora el modelo de calidad, sino que tan solo se especifica cómo debe ser un producto software de calidad. En este apartado están definidas las características y subcaracterísticas de la calidad interna y externa del producto.

2.1.1.2 ISO 9126-2 Métricas externas.

Esta segunda parte del referente ahonda en las métricas externas que permiten medir cuantitativamente las características y subcaracterísticas definidas en el apartado anterior. Las métricas externas se deben usar para medir la calidad del producto software midiendo el comportamiento del sistema del que forma parte. Las métricas externas tan solo pueden ser utilizadas durante las fases de pruebas del ciclo de vida y durante las fases operacionales. La medición se realiza al ejecutar el producto software en el entorno del sistema en el que está destinado a funcionar.

Esta sección incluye la explicación de cómo aplicar las métricas de calidad del software, un conjunto básico de métricas para cada subcaracterística en forma de tablas y un ejemplo de cómo aplicar las métricas durante el ciclo de vida del producto software.

El referente no asigna rangos de valores a las métricas o grado de cumplimiento, puesto que el valor debe ser definido para cada software en particular.

2.1.1.3 ISO 9126-3 Métricas internas.

La tercera parte del referente es similar a la segunda, pero en este caso se centra en las métricas internas, que deben ser utilizadas en un producto software no ejecutable durante sus etapas de desarrollo. Las métricas internas proporcionan a los usuarios la capacidad de medir la calidad de las entregas intermedias y por lo tanto predecir la calidad del producto final. Esto permite al usuario identificar los problemas de calidad e iniciar acciones correctivas lo antes posible en el ciclo de vida del desarrollo.

Esta sección también incluye la explicación de cómo aplicar las métricas de calidad del software, un conjunto básico de métricas para cada subcaracterística en forma de tablas y un ejemplo de cómo aplicar las métricas durante el ciclo de vida del producto software.

El referente recomienda usar métricas internas que tengan relación con las externas, de esta forma pueden ser utilizadas para predecir los valores de las métricas externas.

2.1.1.4 ISO 9126-4 Métricas de calidad en el uso

Esta última parte del referente nos da las métricas de la calidad en el uso. Las características de la calidad en el uso están definidas en el primer apartado de este referente (ISO/IEC 9126-1) y son las siguientes:

- **Eficacia:** La capacidad del producto software de permitir a los usuarios conseguir las metas especificadas con rapidez y completitud, en un contexto específico de uso.
- **Productividad:** La capacidad del producto software de permitir a los usuarios emplear la cantidad apropiada de recursos en relación con la eficacia alcanzada en un contexto de uso especificado.
- **Seguridad:** La capacidad del producto software para lograr niveles aceptables de riesgo de daño a las personas, negocios, software, bienes o el entorno en un contexto de uso especificado.
- **Satisfacción:** La capacidad del producto de software para satisfacer a los usuarios en un contexto de uso especificado.

Las métricas de calidad en el uso miden el grado en el que un producto cumple con las necesidades de los usuarios especificados para conseguir las metas especificadas de eficacia, productividad seguridad y satisfacción. Esto sólo es posible en un entorno de sistema realista.

Las necesidades de calidad del usuario se pueden especificar como requisitos de calidad por las métricas de calidad en el uso, por métricas externas y a veces por métricas internas. Estos requisitos especificados por las métricas deben ser utilizados como criterios cuando se evalúa un producto.

2.1.2 Características definidas por la ISO/IEC 9126

El referente ISO/IEC 9126-1 (Modelo de calidad, primera parte) clasifica la calidad del software en un conjunto estructurado de características y subcaracterísticas: Funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad. Las cuales se detallan a continuación con sus subcaracterísticas.

2.1.2.1 Funcionalidad

La funcionalidad de un producto software es la capacidad de proveer un conjunto de funciones que satisfacen las necesidades implícitas o explícitas cuando el software se usa bajo determinadas condiciones. Sus subcaracterísticas son:

- **Adecuación:** La capacidad del producto software de proveer un conjunto apropiado de funciones para las tareas y objetivos especificados.
- **Exactitud:** La capacidad del producto software de dar los efectos, resultados correctos o acordados con el grado de precisión necesario.
- **Interoperabilidad:** La capacidad del producto software para la interacción con sistemas especificados.
- **Seguridad:** La capacidad del producto software para prevenir el acceso no autorizado ya sea accidental o deliberado, a programas y datos.
- **Cumplimiento de la funcionalidad:** La capacidad del producto software de adherirse a los estándares, convenios, regulaciones o similares relacionados con la funcionalidad.

2.1.2.2 Fiabilidad

La fiabilidad del producto software es la capacidad de mantener su nivel de prestación bajo condiciones establecidas durante un período establecido. Sus subcaracterísticas son:

- Madurez: La capacidad para evitar fallos en el producto por culpa de fallos en el software.
- Tolerancia a fallos: La capacidad del producto software para mantener un nivel especificado de desempeño en casos de fallos de software o de una infracción a su interfaz especificada.
- Recuperabilidad: La capacidad del producto software para restablecer su nivel de desempeño y recuperar los datos directamente afectados en caso de fallo
- Cumplimiento de la Fiabilidad: La capacidad del producto software para adherirse a normas, convenciones o legislación relacionadas con la fiabilidad.

2.1.2.3 Usabilidad

La usabilidad del producto software es la capacidad de ser entendido, aprendido y utilizado por el usuario, además de resultar atractivo para el mismo. Todo esto cuando es utilizado bajo unas condiciones especificadas. Tiene las siguientes subcaracterísticas.

- Comprensión: La capacidad del producto software para permitir al usuario entender de lo que el software es capaz y cómo debe ser utilizado para tareas en concreto y sus condiciones de uso.
- Aprendizaje: La capacidad del producto software para permitir al usuario aprender la aplicación del software
- Operatividad: La capacidad del producto software para permitir al usuario operar y controlar el software.
- Atractividad: La capacidad del producto software de ser atractivo para el usuario.
- Cumplimiento de la usabilidad: La capacidad del producto software para adherirse a normas, convenciones o legislación relacionadas con la usabilidad.

2.1.2.4 Eficiencia

La eficiencia es la capacidad del producto software de proveer el desempeño apropiado a la cantidad de recursos disponibles, bajo condiciones establecidas. Tiene tres subcaracterísticas.

- Comportamiento en el tiempo: La capacidad del producto software para proveer tiempos de respuesta, procesamiento y tasas de rendimiento apropiados.
- Utilización de recursos: La capacidad del producto software para usar las cantidades y tipos de recursos adecuados cuando el software lleva a cabo su función bajo condiciones determinadas.
- Cumplimiento de la eficiencia: La capacidad del producto software para adherirse a normas, convenciones o legislación relacionadas con la eficiencia.

2.1.2.5 Mantenibilidad

La mantenibilidad es la capacidad del producto software de ser modificado. Estas modificaciones incluyen correcciones, mejoras o adaptación del software a cambios en el entorno, los requisitos y las especificaciones funcionales. Sus subcaracterísticas son:

- **Facilidad de análisis:** La capacidad del producto software de ser diagnosticado de deficiencias o causas de fallos en el software. También incluye la identificación de las partes que deben ser modificadas.
- **Facilidad de cambio:** La capacidad del producto software para permitir implementar una modificación específica.
- **Estabilidad:** La capacidad del producto software para evitar efectos no esperados provenientes de las modificaciones del software.
- **Facilidad de pruebas:** La capacidad del producto software para permitir validar el software modificado.
- **Cumplimiento de la mantenibilidad:** La capacidad del producto software para adherirse a normas, convenciones o legislación relacionadas con la mantenibilidad.

2.1.2.6 Portabilidad

En el apartado 6.6 de la ISO/IEC 9126-1 se analizan sin detalle las características de la compatibilidad (Aquí definida como Portability).

Define la portabilidad como “**La capacidad del producto software para ser transferido de un entorno a otro**”.

Y le añade las siguientes subcaracterísticas:

- **Adaptabilidad:** La capacidad del producto software para ser adaptado a diferentes entornos especificados, sin aplicar acciones o mecanismos distintos de aquellos proporcionados para ese propósito por el propio software considerado.
- **Instalabilidad:** La capacidad del producto software para ser instalado en un entorno especificado.
- **Coexistencia:** La capacidad del producto software para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes.
- **Capacidad para reemplazar:** La capacidad del producto software para ser usado en lugar de otro producto software, para el mismo propósito y en el mismo entorno.
- **Cumplimiento de la portabilidad:** La capacidad del producto software para adherirse a normas o convenciones relacionadas con la portabilidad.

Esta definición, junto con estas subcaracterísticas es el punto de partida en este trabajo.

2.2 ISO/IEC 14598

El referente ISO/IEC 14598 establece un marco de trabajo para evaluar la calidad de los productos software proporcionando métricas y requisitos para los procesos de evaluación. Su versión más reciente es del 2005. En particular es utilizada para aplicar los conceptos descritos en la norma ISO/IEC 9126. Se definen y describen las actividades necesarias para analizar los requisitos de evaluación, para especificar, diseñar y realizar acciones de evaluación y para concluir la evaluación de cualquier tipo de producto de software.

La norma define las principales características del proceso de evaluación:

- **Repetitividad:** La evaluación repetida de un mismo producto, realizada bajo la misma especificación y con el mismo evaluador, debe producir resultados que pueden aceptarse como idénticos.

- Reproducibilidad: La evaluación del mismo producto, con la misma especificación de evaluación, pero ejecutada por un evaluador diferente, debe producir resultados que pueden aceptarse como idénticos.
- Imparcialidad: La evaluación no debe ser influenciada en comparación con cualquier otro resultado en particular.
- Objetividad: Los resultados de la evaluación deben ser verdaderos, y no ser influenciados por las opiniones o sentimientos del evaluador.

Para estas características se describen las siguientes medidas concretas:

- Análisis de los requisitos de evaluación.
- Evaluación de las especificaciones.
- Evaluación del diseño y definición del plan de evaluación.
- Ejecución del plan de evaluación.
- Evaluación de la conclusión.

El referente está dividido en seis partes:

- ISO/IEC 14598-1 Vista general: Provee una visión general de las otras cinco partes y explica la relación entre la evaluación del producto software y el modelo de calidad definido en la ISO/IEC 9126
- ISO/IEC 14598-2 Planificación y administración: Contiene requisitos y guías para las funciones de soporte tales como la planificación y gestión de la evaluación del producto del software.
- ISO/IEC 14598-3 Proceso para desarrolladores: Provee los requisitos y guías para la evaluación del producto software cuando la evaluación es llevada a cabo en paralelo con el desarrollo por parte del desarrollador.
- ISO/IEC 14598-4 Proceso para clientes (adquisidores): Provee los requisitos y guías para que la evaluación del producto software sea llevada a cabo en función a los compradores que planean adquirir o reutilizar un producto de software existente o pre-desarrollado.
- ISO/IEC 14598-5 Proceso para evaluadores: Proceso para avaladores: provee los requisitos y guías para la evaluación del producto software cuando la evaluación es llevada a cabo por evaluadores independientes.
- ISO/IEC 14598-6 Documentación de módulos y evaluación: provee las guías para la documentación del módulo de evaluación.

La portabilidad aparece en las partes 4 (proceso para clientes) y 5 (proceso para evaluadores) del referente. En el primer caso se trata de métricas y en el segundo de técnicas de evaluación

2.2.1 ISO/IEC 14598-4

Según la ISO/IEC 14598-1 (Vista general) el proceso de evaluación tiene 4 pasos:

- Establecer los requisitos de evaluación.
- Especificar la evaluación.

- Diseñar la evaluación.
- Ejecutar la evaluación

La portabilidad se menciona como una métrica en el segundo paso del proceso de evaluación (especificar la evaluación).

Estas métricas se detallan en el apéndice B, que es de tipo informativo y pone unas orientaciones para los valores del umbral de aceptación, que deben estar siempre basados en la experiencia propia y servir como orientación, no como reglas inamovibles.

La portabilidad es tenida en cuenta tanto como métrica externa, como métrica en la calidad de uso.

2.2.1.1 ISO/IEC 14598-4: La portabilidad como métrica externa

La portabilidad aparece primero como una métrica externa. Estas métricas sirven para correlacionar aspectos medibles con la calidad del software.

La tabla B.2 que aparece en este apartado del referente nos indica la prioridad de las características a medir teniendo como objetivo la integridad del software.

Objetivo integridad software	Prioridad de características de calidad	Subcaracterística seleccionada	Métrica externa seleccionada	Posible criterio de aceptación
BAJA	1. Funcionalidad	Precisión	Número de resultados precisos frente a resultados esperados precalculados.	95%
	2. Usabilidad	Operabilidad	El número de mensajes claros evaluados recibido en frente al total de mensajes revisado.	80%
	3. Portabilidad	Instalabilidad	Número de módulos que se vuelve a compilar frente al número total de módulos cuando se transfiere a una nueva plataforma.	<6 módulos
	4. Eficiencia	Comportamiento en el tiempo	Tiempo transcurrido entre el inicio de la acción del sistema y la recepción de la respuesta del sistema.	<5 segundos
	5. Fiabilidad	Tolerancia a fallos	Número de fallos para detectar la entrada errónea frente al número de entradas erróneas introducidas.	25%
	6. Mantenibilidad (no requerido)	-no requerido-	-no requerido-	-
ALTA	1. Fiabilidad	Disponibilidad	Tiempo medio entre fallos de un período de funcionamiento especificado.	>6 meses
	2. Funcionalidad	Idoneidad	El número de requisitos obligatorios reunidos en la especificación de requisitos del software frente al número total de requisitos obligatorios.	100%
	3. Mantenibilidad	Cambiabilidad	El número de módulos que necesitan ser cambiados para identificar probables cambios.	1
	4. Eficiencia	Utilización de recursos	El porcentaje de carga de la CPU durante un período de funcionamiento especificado en el caso de peores condiciones.	80%
	5. Usabilidad	Inteligibilidad	La cantidad de tiempo que se requiere para los usuarios especificados para aprender a utilizar el software para producir los resultados especificados.	<10 minutos
	Portabilidad (no requerido)	-no requerido-	-no requerido-	-

Tabla B.2: Especificación de las características, subcaracterísticas y métricas externas de la calidad del software, con ejemplos

Si nuestro objetivo es un software de integridad baja (en el que el acceso a los datos no haría falta proteger apenas), la portabilidad está en tercer lugar (después de funcionalidad y usabilidad) y la subcaracterística que da como ejemplo es la instalabilidad. Propone medir el número de módulos que

necesitan ser recompilados cuando transferimos el software a una nueva plataforma, el número orientativo de módulos debería ser 6.

Por el contrario, si nuestro objetivo en un software de alta integridad, la portabilidad es la característica menos importante y ni siquiera se ponen ejemplos.

2.2.1.2 ISO/IEC 14598-4: La portabilidad como métrica en la calidad de uso

La portabilidad también aparece como una métrica en la calidad de uso, las cuales correlacionan la visión del usuario respecto a la calidad del sistema con parámetros medibles.

La tabla B.3 nos detalla las métricas en la calidad de uso y las clasifica según el tipo de usuario.

Usuario	Características objetivo de calidad externa	Atributos de calidad en el uso	Ejemplo de medida de calidad en el uso	Criterio de aceptación
Operador, usuario final	Funcionalidad, fiabilidad, usabilidad, eficiencia	Eficacia (en relación con los objetivos de utilización del sistema / producto con la exactitud y exhaustividad con la que estos objetivos pueden ser alcanzados)	- Número de errores cometidos en la interpretación de resultados en la pantalla - Número de desviaciones de un formato especificado en la transcripción de un documento	- Cuanto más cercano a cero, mejor - Cuanto más cercano a cero, mejor
		Eficiencia (nivel de eficacia alcanzada en relación con el gasto mental o esfuerzo físico).	- Tiempo de trabajo y los costes laborales del tiempo del usuario - Coste de los recursos y el equipo utilizado - Coste de cualquier entrenamiento requerido por el usuario	- Cuanto más bajo, mejor - Cuanto más bajo, mejor - Cuanto más bajo, mejor
		Satisfacción (confort y aceptabilidad de uso)	- Cuestionarios de satisfacción de interacción con el usuario. - Ratio: Comentarios positivos frente a negativos durante la utilización. - Cuestionarios de carga de trabajo cognitivo.	- Cuanto más alto, mejor - Cuanto más alto, mejor - Media de población por tipo de usuario
Mantenedor, instalador	Mantenibilidad, portabilidad	Eficacia (en relación con los objetivos de utilización del sistema / producto con la exactitud y exhaustividad con la que estos objetivos pueden ser alcanzados)	- Número de módulos afectados por cada solicitud de cambio. - Número de instalaciones exitosas frente al número de intentos de instalación.	- Cuanto más bajo, mejor - Cuanto más cercano a 1.0, mejor
		Eficiencia (nivel de eficacia alcanzada en relación con el gasto mental o esfuerzo físico).	- Tiempo de trabajo y los costes laborales del tiempo del usuario - Coste de los recursos y el equipo utilizado - Coste de cualquier entrenamiento requerido por el usuario	- Cuanto más bajo, mejor - Cuanto más bajo, mejor - Cuanto más bajo, mejor
		Satisfacción (confort y aceptabilidad de uso)	- Cuestionarios de satisfacción de interacción con el usuario. - Ratio: Comentarios positivos frente a negativos durante la utilización. - Cuestionarios de carga de trabajo cognitivo.	- Cuanto más alto, mejor - Cuanto más alto, mejor - Media de población por tipo de usuario

Tabla B.3: Especificación de las métricas de calidad de uso, con ejemplos

Las referidas a la portabilidad se orientan al usuario que debe mantener e instalar el software y se presentan siempre junto con las de mantenibilidad. Son las siguientes:

- Efectividad (relativa a los objetivos de utilización del sistema/producto con la exactitud y exhaustividad con la que estos objetivos pueden ser alcanzados)
 - Número de módulos afectados por cada modificación (cuantos menos mejor)
 - Número de instalaciones exitosas entre el número de intentos (lo más cercano a 1.0)
- Eficiencia (nivel de efectividad alcanzada en relación con el esfuerzo físico o mental)
 - Tiempo de tarea y costes laborales del usuario (cuantos menos mejor)
 - Coste de los recursos y equipamiento usado (cuantos menos mejor)
 - Coste de cualquier entrenamiento que requiera el usuario (cuantos menos mejor)
- Satisfacción (confort y aceptabilidad durante la utilización)
 - Cuestionarios de satisfacción de la interacción con el usuario (cuanto mayor mejor)
 - Ratio de comentarios positivos vs negativos durante la utilización (cuanto mayor mejor)
 - Cuestionarios de carga de trabajo cognitiva (en la media de la población según el tipo de usuario)

2.2.1.3 ISO/IEC 14598-4: La portabilidad como método de evaluación

Por último la portabilidad también aparece en la tabla B.4, la cual nos muestra un ejemplo de cómo clasificar los métodos de evaluación de las características de calidad de software según la relación costes-eficacia.

Método de evaluación	Ranking de coste	Ranking de efectividad					
		Funcionalidad	Fiabilidad	Usabilidad	Eficiencia	Mantenibilidad	Portabilidad
Documentos del producto final, cursos y entrenamiento	Bajo	Alto	Bajo	Alto	Medio	Bajo	Alto
Productos intermedios	Alto	Bajo	Medio	Bajo	Medio	Bajo	Alto
Historial de operaciones - Proveedor	Medio	Medio	Alto	Bajo	Bajo	Medio	Medio
Historial de operaciones - Cliente	Medio	Alto	Medio	Alto	Medio	Alto	Alto

Tabla B.4: Ranking coste-eficiencia de métodos de evaluación

En lo referente a la portabilidad, clasifica como alta los métodos de evaluación según la documentación de los productos finales, cursos y entrenamiento. También considera alta la evaluación de los productos intermedios y del historial de acciones del cliente. Por el contrario califica de medio la evaluación del historial de acciones del proveedor.

2.2.2 ISO/IEC 14598-5

La portabilidad también aparece en la **ISO/IEC 14598-5** (Proceso para evaluadores) como parte de las técnicas de evaluación, de cada una de las características de calidad de software según la ISO/IEC 9126 (Funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad). De cada característica da 3 métricas, ordenadas de menor a mayor según el nivel de exigencia, esta escala mide la exigencia que debe tener un producto conforme a los criterios de seguridad, protección, económicos y medioambientales. Para la portabilidad nos da una escala de:

- Análisis de los procedimientos de instalación del software. (Menos exigente)

- Programar reglas de verificación.
- Análisis del diseño del software. (Más exigente)

2.3 ISO/IEC 25000

ISO/IEC 25000 es una familia de estándares que reemplaza a la ISO/IEC 9126 y a la ISO/IEC 14598. Su objetivo es organizar, enriquecer y unificar las series que cubren dos procesos principales: Especificación de requisitos de calidad del software y evaluación de la calidad del software, soportada por el proceso de medición de calidad del software.

A esta familia también se la conoce por “Software Product **Quality Requirements and Evaluation**” y más comúnmente por sus siglas SQuaRE.

Este referente, al actualizar la ISO/IEC 9126 y la ISO/IEC 14598, no tiene previsto proveer más métricas de las que ya vienen en la 9126. Más bien es un intento de unificación para resolver las incoherencias entre los dos referentes anteriormente mencionados.

Dentro de la familia se han publicado los siguientes referentes de los siguientes referentes:

- **ISO/IEC 25000 Guía para SQuaRE:** Es tan sólo una guía de SQuaRE, describe la arquitectura, la terminología y los modelos de referencia.
- **ISO/IEC 25001 Planificación y gestión:** Describe el soporte necesario para la gestión en la evaluación y especificación de requisitos de calidad.
- **ISO/IEC 25010 Modelo de calidad y guía:** Define un modelo de calidad para el producto software. Se detalla en el siguiente punto.
- **ISO/IEC 25012 Modelo de calidad de datos:** Define el modelo de calidad de datos para definir la evaluación de un producto de datos. La Calidad del Producto de Datos se puede entender como el grado en que los datos satisfacen los requisitos definidos por la organización a la que pertenece el producto. Se le asignan 15 características: Precisión, completitud, consistencia, credibilidad, actualidad, accesibilidad, conformidad, confidencialidad, eficiencia, exactitud, trazabilidad, comprensibilidad, disponibilidad, portabilidad y recuperabilidad.
- **ISO/IEC 25020 Modelo de referencia para la medida y guía:** Presenta una guía y un modelo para realizar la medición de la calidad del software. Modelo de referencia para la medida con guía: presenta una introducción y un modelo de referencia común para las primitivas, medidas de calidad interna, externa y en el uso. En la guía se proporciona ayuda para seleccionar un conjunto adecuado de medidas de los estándares internacionales ISO/IEC 9126-1 a 9126-4 e ISO/IEC 14598-1
- **ISO/IEC 25021 Elementos de medida de la calidad:** Define y especifica medidas base y derivadas a utilizar durante el desarrollo del software. Este conjunto de medidas se utilizará como entrada en el proceso de medida de la calidad interna, externa y en el uso.
- **ISO/IEC 25022 Medidas de calidad interna:** define las medidas de calidad interna en función de características y sub características.
- **ISO/IEC 25023 Medidas de calidad externa:** define las medidas de calidad externa en función de características y subcaracterísticas.

- **ISO/IEC ISO/IEC 25024 Medidas de calidad en uso:** describe un conjunto de medidas para la calidad en el uso y una guía para la utilización de estas medidas
- **ISO/IEC 25030 Requisitos de calidad:** Contiene recomendaciones para realizar la especificación de los requisitos de calidad del producto software.
- **ISO/IEC 25040 Modelo de referencia para la evaluación:** Define el proceso para llevar a cabo la evaluación del producto software. Se detalla en el punto 2.3.1
- **ISO/IEC 25041 Módulos de evaluación:** describe la estructura y contenido de la documentación que debe describir los módulos de evaluación.
- **ISO/IEC 25042 Proceso de evaluación para desarrolladores:** proporciona requisitos y recomendaciones de carácter práctico para la implementación de la evaluación cuando esta se da en paralelo con el desarrollo.
- **ISO/IEC 25043 Proceso de evaluación para compradores:** describe requisitos y recomendaciones para la medida y evaluación sistemática de productos software comercial, productos desarrollados a medida, o productos a modificar bajo contrato.
- **ISO/IEC 25044 Proceso de evaluación para evaluadores:** detalla requisitos y recomendaciones para la evaluación de software de forma que dicha evaluación sea fiable y comprensible

A nosotros nos interesan la 25010 y la 25040, las cuales analizaremos a continuación. Nos interesan debido a que son las que actualizan la ISO/IEC 9126 y la ISO/IEC 14598.

2.3.1 ISO/IEC 25010

En el referente **ISO/IEC 25010** se define un modelo de calidad del producto software compuesto por las siguientes ocho características:

- **Adecuación funcional:** Representa la capacidad del producto software para proporcionar funciones que satisfacen las necesidades declaradas e implícitas, cuando el producto se usa en las condiciones especificadas.
- **Fiabilidad:** Capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados.
- **Eficiencia de desempeño:** Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones.
- **Usabilidad:** Capacidad del producto software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa bajo determinadas condiciones.
- **Seguridad:** Capacidad de protección de la información y los datos de manera que personas o sistemas no autorizados no puedan leerlos o modificarlos.
- **Compatibilidad:** Capacidad de dos o más sistemas o componentes para intercambiar información y/o llevar a cabo sus funciones requeridas cuando comparten el mismo entorno hardware o software.
- **Mantenibilidad:** Esta característica representa la capacidad del producto software para ser modificado efectiva y eficientemente, debido a necesidades evolutivas, correctivas o perfectivas.
- **Transferibilidad:** Capacidad del producto o componente de ser transferido de forma efectiva y eficiente de un entorno hardware, software, operacional o de utilización a otro.

Como podemos ver el concepto de portabilidad definido en ISO/IEC 9126-1 se ha desdoblado en compatibilidad y en transferibilidad.

Este referente define la compatibilidad como “la capacidad de dos o más componentes software de intercambiar información y/o realizar sus funciones asignadas mientras comparten el mismo entorno hardware o software”. Y le da las siguientes subcaracterísticas:

- Reemplazabilidad: El grado en que el producto software puede ser utilizado en lugar de otro producto software específico para el mismo propósito y en el mismo entorno.
- Coexistencia: El grado en el que el producto software puede coexistir con otro software independiente en un entorno común compartiendo recursos comunes sin ningún impacto negativo.
- Interoperabilidad: El grado en que el producto software puede utilizado cooperativamente por uno o más productos software.

Por otro lado este referente define la transferibilidad como “el grado con el que el producto software puede ser transferido de un entorno a otro” y le asigna las siguientes características:

- Portabilidad: La facilidad con la que un sistema o componente pueden ser transferidos de un entorno hardware o software a otro.
- Adaptabilidad: El grado en que el producto de software puede ser adaptado para diferentes entornos específicos sin aplicar acciones o medios distintos de los previstos para este propósito en el software considerado.
- Instalabilidad: El grado en que el producto software puede ser exitosamente instalado y desinstalado en un entorno específico.

Por lo que vemos que el referente ISO/IEC 25010 analiza sin detalle las características de la portabilidad y las subdivide en compatibilidad (Reemplazabilidad, coexistencia e interoperabilidad) y transferibilidad (Portabilidad, adaptabilidad, instalabilidad).

2.3.1 ISO/IEC 25040

El ISO/IEC 25040 define el proceso para llevar a cabo la evaluación del producto software. Está destinado a sustituir la ISO/IEC 14598. Dicho proceso de evaluación consta de un total de cinco actividades (frente a las cuatro de la 14598, se añade una quinta actividad: Concluir la evaluación). Cada actividad tiene asignadas unas tareas concretas.

Actividad 1: Establecer los requisitos de la evaluación: El primer paso del proceso de evaluación consiste en establecer los requisitos de la evaluación. Consta de 4 tareas:

- **Tarea 1.1: Establecer el propósito de la evaluación:** En esta tarea se documenta el propósito por el que la organización quiere evaluar la calidad de su producto software (asegurar la calidad del producto, decidir si se acepta un producto, determinar la viabilidad

del proyecto en desarrollo, comparar la calidad del producto con productos de la competencia, etc.).

- **Tarea 1.2: Obtener los requisitos de calidad del producto:** En esta tarea se identifican las partes interesadas en el producto software (desarrolladores, posibles adquirientes, usuarios, proveedores, etc.) y se especifican los requisitos de calidad del producto utilizando un determinado modelo de calidad.
- **Tarea 1.3: Identificar las partes del producto que se deben evaluar:** Se deben identificar y documentar las partes del producto software incluidas en la evaluación. El tipo de producto a evaluar (especificación de requisitos, diagramas de diseño, documentación de las pruebas, etc.) depende de la fase en el ciclo de vida en que se realiza la evaluación y del propósito de ésta.
- **Tarea 1.4: Definir el rigor de la evaluación:** Se debe definir el rigor de la evaluación en función del propósito y el uso previsto del producto software, basándose, por ejemplo, en aspectos como el riesgo para la seguridad, el riesgo económico o el riesgo ambiental. En función del rigor se podrá establecer qué técnicas se aplican y qué resultados se esperan de la evaluación.

Actividad 2: Especificar la evaluación: En esta actividad se especifican los módulos de evaluación (compuestos por las métricas, herramientas y técnicas de medición) y los criterios de decisión que se aplicarán en la evaluación. Consta de 3 tareas

- **Tarea 2.1: Seleccionar los módulos de evaluación:** En esta tarea el evaluador selecciona las métricas de calidad, técnicas y herramientas (módulos de evaluación) que cubran todos los requisitos de la evaluación. Dichas métricas deben permitir que, en función de su valor, se puedan realizar comparaciones fiables con criterios que permitan tomar decisiones. Para ello se puede tener en cuenta la Norma ISO/IEC 25020.
- **Tarea 2.2: Definir los criterios de decisión para las métricas:** Se deben definir los criterios de decisión para las métricas seleccionadas. Dichos criterios son umbrales numéricos que se pueden relacionar con los requisitos de calidad y posteriormente con los criterios de evaluación para decidir la calidad del producto. Estos umbrales se pueden establecer a partir de *benchmarks*, límites de control estadísticos, datos históricos, requisitos del cliente, etc.
- **Tarea 2.3: Definir los criterios de decisión de la evaluación:** Se deben definir criterios para las diferentes características evaluadas a partir de las subcaracterísticas y métricas de calidad. Estos resultados a mayor nivel de abstracción permiten realizar la valoración de la calidad del producto software de forma general.

Actividad 3: Diseñar la evaluación: En esta actividad se define el plan con las actividades de evaluación que se deben realizar. Tan sólo tiene una tarea.

- **Tarea 3.1: Planificar las actividades de la evaluación:** Se deben planificar las actividades de la evaluación teniendo en cuenta la disponibilidad de los recursos, tanto humanos como materiales, que puedan ser necesarios. En la planificación se debe tener en cuenta el presupuesto, los métodos de evaluación y estándares adaptados, las herramientas de evaluación, etc. El plan de evaluación se revisará y actualizará proporcionando información adicional según sea necesario durante el proceso de evaluación.

Actividad 4: Ejecutar la evaluación: En esta actividad se ejecutan las actividades de evaluación obteniendo las métricas de calidad y aplicando los criterios de evaluación. Se compone de tres tareas:

- **Tarea 4.1: Realizar las mediciones:** Se deben realizar las mediciones sobre el producto software y sus componentes para obtener los valores de las métricas seleccionadas e indicadas en el plan de evaluación. Todos los resultados obtenidos deberán ser debidamente registrados.
- **Tarea 4.2: Aplicar los criterios de decisión para las métricas:** Se aplican los criterios de decisión para las métricas seleccionadas sobre los valores obtenidos en la medición del producto.
- **Tarea 4.3: Aplicar los criterios de decisión de la evaluación:** En esta última tarea se deben aplicar los criterios de decisión a nivel de características y subcaracterísticas de calidad, produciendo como resultado la valoración del grado en que el producto software cumple los requisitos de calidad establecidos.

Actividad 5: Concluir la evaluación: En esta actividad se concluye la evaluación de la calidad del producto software, realizando el informe de resultados que se entregará al cliente y revisando con éste los resultados obtenidos. Las tareas a realizar son cuatro

- **Tarea 5.1: Revisar los resultados de la evaluación:** Mediante esta tarea, el evaluador y el cliente de la evaluación (en caso de existir) realizan una revisión conjunta de los resultados obtenidos, con el objetivo de realizar una mejor interpretación de la evaluación y una mejor detección de errores.
- **Tarea 5.2: Crear el informe de evaluación:** Una vez revisados los resultados, se elabora el informe de evaluación, con los requisitos de la evaluación, los resultados, las limitaciones y restricciones, el personal evaluador, etc.
- **Tarea 5.3: Revisar la calidad de la evaluación y obtener *feedback*:** El evaluador revisará los resultados de la evaluación y la validez del proceso de evaluación, de los indicadores y de las métricas aplicadas. El *feedback* de la revisión debe servir para mejorar el proceso de evaluación de la organización y las técnicas de evaluación utilizadas.
- **Tarea 5.4: Tratar los datos de la evaluación:** Una vez finalizada la evaluación, el evaluador debe realizar el adecuado tratamiento con los datos y los objetos de la evaluación según lo acordado con el cliente (en caso de ser una tercera parte), devolviéndolos, archivándolos o eliminándolos según corresponda.

3. Definición de un referente de auditoría

3.1 Introducción

En este capítulo se analizan con detenimiento las métricas internas y externas que nos ayudan a medir la calidad de un producto software. Se comienza con el análisis detallado de las métricas externas e internas que aparecen en la ISO/IEC 9126-2 y en la ISO/IEC 9126-3 respectivamente. Después se proponen unas nuevas métricas para completar y complementar las anteriormente vistas. Finalmente se reúnen todas las métricas de este capítulo en una completa tabla resumen.

El análisis y explicación teórico realizado y expuesto en este TFG para cada una de las métricas, se apoya en ejemplos y supuestos prácticos con el objetivo de una mejor comprensión de cada una de las métricas por parte del lector.

3.2 Métricas externas de portabilidad definidas en la ISO 9126-2²

El estándar nos indica que las métricas externas de portabilidad deben ser capaces de medir atributos tales como el comportamiento del usuario o del sistema durante las actividades de portabilidad, es decir, miden el comportamiento del usuario o del sistema mientras se está realizando la portabilidad del producto software de un entorno a otro. Recordemos que las métricas externas se deben usar para medir la calidad del producto software midiendo el comportamiento del sistema del que forma parte. Las métricas externas tan solo pueden ser utilizadas durante las fases de pruebas del ciclo de vida y durante las fases operacionales (mientras se está utilizando el producto software por un usuario). La medición se realiza al ejecutar el producto software en el entorno del sistema en el que está destinado a funcionar.

3.2.1 Métricas externas de adaptabilidad

Estas métricas deben ser capaces de medir atributos tales como el comportamiento del sistema o del usuario que está realizando la adaptación del producto software en diferentes entornos. Cuando un usuario tiene que realizar un procedimiento de adaptación distinto al que previamente le proporcionaba el software para una necesidad de adaptación específica, es necesario medir el esfuerzo necesario para la adaptación por parte del usuario.

El método de evaluación de estas métricas se realiza mediante la observación del comportamiento de la persona que intenta adaptar el software, sea el usuario o la persona encargada de su mantenimiento. Después se aplica una fórmula para obtener un valor (normalmente entre 0 y 1) que nos dé una idea del grado de cumplimiento de la métrica en concreto. La fórmula a aplicar se especifica en cada una de las métricas.

Los datos a analizar en estas métricas se suelen encontrar en el informe de resolución de problemas y en el informe de operaciones. Todas estas métricas están dirigidas a desarrolladores, encargados de mantenimiento y operadores del producto final.

² Más concretamente en el apartado 6.8

3.2.1.1 Adaptabilidad de estructuras de datos

Miden si el usuario o el encargado de mantenimiento pueden adaptar de manera sencilla el software a los nuevos tipos de datos en el nuevo entorno. Se mide mediante la fórmula $X = A / B$, siendo A el número de estructuras de datos que son operables (otras funciones del mismo programa pueden utilizarlas) pero no son observables debido a limitaciones en la adaptabilidad y B el número de datos que se espera que sean operables en el nuevo entorno. X es un coeficiente entre 0 y 1, un valor más cercano a 1 es mejor. Las estructuras de datos que cuentan tanto A como B deben referirse siempre al mismo tipo de datos.

Por ejemplo, queremos transferir a otro entorno un programa que contiene una base de datos de usuario compuesto por tuplas formadas por email, una contraseña y una foto. Si al portarlo a otro entorno el usuario es capaz de trabajar con esos datos pero no es capaz de acceder a la información en sí (es decir, la base de datos se comporta como una caja negra), entonces esta estructura de datos se sumaría al conteo de B, pero no al de A.

3.2.1.2 Adaptabilidad al entorno Hardware

Esta métrica mide la facilidad con la que un usuario o un mantenedor adaptan el software al entorno. También mide cómo el software es capaz de adaptarse por sí mismo al nuevo entorno. La fórmula a aplicar es la siguiente: $X = 1 - A / B$, siendo A el número de funciones operativas cuyas tareas no han sido completadas de manera totalmente satisfactoria durante el testeo de software en el entorno hardware. La B es el número de funciones que se han examinado. Cuanto mayor sea el valor de la X, más adaptable resultará el producto software.

Por ejemplo, si al transferir un software, la parte encargada de girar imágenes (suponiendo que no sea un cuello de botella) tarda más tiempo del que debería, convirtiéndose en un nuevo cuello de botella y este retraso es producido por la memoria (un componente hardware), La función operativa “girar imagen” se incluiría tanto en el conteo de A como en el de B.

3.2.1.3 Adaptabilidad al entorno organizacional

Esta métrica mide cómo de adaptable es el software a un cambio organizacional, está relacionada con las operaciones de negocio del usuario. En este caso la fórmula vuelve a ser $X = 1 - A / B$, siendo A el número de funciones implementadas que no son capaces de adaptarse a múltiples organizaciones y entornos de negocio. B es el total de funciones analizadas para esta métrica. Cuanto mayor sea el valor de X, mejor.

Por ejemplo, si en una empresa que antes tenía un solo usuario con el rol de administrador, se decide que debe haber más de uno, y debido a esto se producen problemas de coherencia en el software, la función “administrador” no es capaz de adaptarse a distintas organizaciones, y por tanto se incluiría en el conteo de A y B.

3.2.1.4 Amigabilidad con el usuario para la portabilidad

Esta métrica es la encargada de medir el esfuerzo invertido por el usuario realizando operaciones de portabilidad sobre el producto. La fórmula en este caso es la suma del tiempo invertido en completar la adaptación del software al entorno del usuario cuando este intenta instalar o configurar el software. Un valor de T bajo sería lo más deseable.

Por ejemplo, en dos productos software (X y Z) de complejidad similar que se han portado a otro entorno se ha invertido un total de 140 horas en el programa X y un total de 47 horas en el programa Z, por lo tanto el valor de T sería de 140 en el programa X y de 47 en el programa Z. Entonces en este aspecto el programa Z sería más portable.

3.2.1.5 Adaptabilidad al entorno del sistema software

En esta métrica se mide la adaptabilidad del producto software al sistema operativo o software de red. Cuantifica cuan adaptable es el software a los cambios en el entorno relacionados con el sistema software. La fórmula vuelve a ser $X = A / B$, siendo A el número de funciones relacionadas con esta métrica que se adaptan al sistema software y B el número total de funciones evaluadas

Por ejemplo, se ha portado una aplicación que entre otras cosas escanea huellas dactilares a un nuevo entorno, en el nuevo entorno la aplicación no es capaz de acceder al lector de huellas por un problema con el driver, por tanto la función “escanear huella” se incluiría en el conteo de B, pero no en el de A

3.2.2 Métricas externas de instalabilidad

Estas métricas deben medir atributos sobre el sistema o el usuario que está intentando instalar el software en un entorno específico de usuario final. Intentan responder a la pregunta: ¿El usuario o la persona encargada del mantenimiento puede instalar el software en el entorno de operaciones? El método a seguir es la observación durante el proceso de instalación. Estas métricas también están orientadas hacia desarrolladores, encargados de mantenimiento y operadores del producto final. Y se nutren de los datos recogidos en el informe de resolución de problemas y en el informe de operaciones.

3.2.2.1 Facilidad de Instalación

Esta métrica mide cuan fácil es para un usuario instalar por primera vez el software en un entorno nuevo. La fórmula a aplicar es $X = A / B$ siendo A el número de veces que el usuario ha tenido éxito en cambiar la instalación a su conveniencia. B es el número de veces que el usuario lo ha intentado, hayan tenido éxito o no. Un valor de X más cercano a 1 es lo más deseable.

Por ejemplo, un usuario quiere instalar una suite ofimática. En el primer intento el asistente de instalación se ha bloqueado y se ha cerrado inesperadamente, en el segundo intento lo ha conseguido instalar, pero se le ha olvidado instalar los *plugins* del pdf. A la tercera lo instala tal y como él quiere. El valor de X sería 1/3, ya que lo ha intentado 3 veces y lo ha conseguido sólo una. Suponiendo que este usuario tiene que instalar el mismo programa en 15 ordenadores más y lo consigue a la primera, el valor de X sería 16/18, ya que ha conseguido 16 instalaciones exitosas frente a los 18 intentos (16 exitosos y 2 erróneos)

3.2.2.2 Facilidad para reintentar la configuración

Esta métrica mide si el usuario es capaz de reinstalar y configurar el software. La fórmula en este caso es $X = 1 - A / B$, siendo A el número de fallos del usuario al intentar reinstalar el software y B el número de intentos. Un valor de X más cercano a 1 es lo más deseable.

Suponiendo el caso del apartado anterior, si este usuario ha intentado reinstalar el software ofimático en los 16 ordenadores y lo ha conseguido a la primera, el valor de X sería 16/16, o lo que es lo mismo, 1 (el valor más deseable)

3.2.3 Métricas externas de coexistencia

Estas métricas deben medir atributos el comportamiento del sistema en el cual el usuario intenta usar el software a evaluar con otro software independiente en un entorno común y compartiendo recursos. Una vez más los datos a observar deben sacarse del informe de resolución de problemas y en el informe de operaciones. En este referente tan solo nos encontramos con una métrica, que es un ratio; para la medición de esta métrica debe usarse el software a evaluar junto con otros softwares que el usuario use a menudo.

3.2.3.1 Posibilidad de coexistencia

Para esta métrica se emplea la fórmula $X = A/T$, siendo A el número de restricciones o fallos inesperados que el usuario encuentra durante el uso del software de manera concurrente con otro software. T es el tiempo que se emplean de manera simultánea el software a evaluar y otros de frecuente uso. Cuanto más cercano a X sea el valor de 0, mejor.

Por ejemplo, si durante una evaluación de 7 horas de una aplicación de retoque fotográfico que se ejecuta concurrentemente con otra aplicación de retoque fotográfico (como Adobe Photoshop) se han producido 3 errores de desbordamiento de pila y 2 reinicios inesperados de la aplicación, el valor de X sería 5/7.

3.2.4 Métricas externas de reemplazabilidad

Estas métricas miden el comportamiento del sistema o del usuario que está intentando utilizar el software en lugar de otro software específico, todo esto en el mismo entorno. Para evaluar esta métrica tomaremos los datos del informe de resolución de problemas y del informe de operaciones. El método de medición será la observación del comportamiento del usuario mientras está reemplazando el software por otro previamente instalado. Esta métrica implica el trabajo de desarrolladores, encargados de mantenimiento y operadores, y se puede aplicar tanto en el caso de reemplazar completamente el software por uno nuevo, como para una actualización del software.

3.2.4.1 Uso continuado de los datos

Esta métrica intenta medir si el usuario o el “mantenedor” son capaces de seguir usando los mismos datos que usaba con el software anterior después de haber sido reemplazado por uno nuevo. La fórmula a aplicar es $X=A/B$, siendo A el número de datos que se usaban en el software anterior y que se pueden seguir utilizando en el nuevo software. B sería el número de datos que se usaban en el software anterior y de los cuales se planea su reutilización en el nuevo software. Un valor de X mayor es más deseable.

Por ejemplo, un producto software de gestión hospitalaria utilizaba datos de tres bases de datos, dos de ellas de tipo no relacional y otra de tipo relacional, concretamente SQL, si al reemplazar el software por uno nuevo este no es capaz de leer los datos de la base de datos SQL porque le falta el conector, la cantidad de datos que había en la base de datos no relacional se suma al conteo de B, pero no al de A. Por lo tanto el valor de la X disminuye, lo cual es indicador de que el producto es

menos reemplazable, ya que los datos que faltan habría que obtenerlos de otras fuentes o por otros métodos no previstos por el usuario del producto.

3.2.4.2 Inclusividad de funciones

Esta métrica nos indica la facilidad con la que un usuario o mantenedor puede continuar utilizando funciones similares después de reemplazar el software. La fórmula a aplicar es $X = A/B$, siendo A el número de funciones que producen resultados similares a los producidos por el software anterior y que no necesitan ser cambiadas. B es el número de funciones similares a las funciones del software a reemplazar. X debería ser el mayor valor posible.

En el ejemplo del remplazamiento de la aplicación de retoque fotográfico si las nuevas funciones “girar imagen”, “convertir a blanco y negro” y “recortar” funcionan como se espera pero la función “espejar” tan sólo gira la imagen en vez de mostrarla como reflejada en un espejo, el valor de X sería $\frac{3}{4}$. Ya que de 4 funciones, tan solo tres funcionan correctamente.

3.2.4.3 Consistencia funcional al soporte del usuario

Esta métrica mide cómo de consistentes son los nuevos componentes con los ya existentes en la interfaz de usuario. Para la medición de esta métrica además de observar el comportamiento del usuario, también se le debe preguntar su opinión. La fórmula es la siguiente: $X = 1 - A / B$. Siendo A el número de nuevas funciones que el usuario encuentra inaceptables e inconsistentes con sus expectativas. B es el número de nuevas funciones. Un valor de X mayor es lo más deseable. Esta métrica está destinada al usuario final, al diseñador de la interfaz de usuario, al mantenedor, al desarrollador y al SQA (Software Quality Assurance)

Por ejemplo, si a un producto software se le realiza una actualización en la que se añade una nueva barra de herramientas que el usuario considera inconsistente con el resto de elementos de la interfaz porque no tiene la misma distribución que el resto de barras de herramientas, la función “nueva barra de herramientas” se incluiría tanto en el conteo de A como en el de B

3.2.5 Métricas externas de cumplimiento de la portabilidad

Estas métricas miden el número de fallos de cumplimiento de la portabilidad en el software, en cuanto este no se adhiere a los estándares, convenios o regulaciones relativos a la portabilidad. Tan solo existe una métrica, la cual afecta al usuario final y al proveedor del software.

3.2.5.1 Cumplimiento de la portabilidad

Esta métrica mide cómo de conforme es la portabilidad del producto con regulaciones, estándares y convenciones aplicables. El método a aplicar es contar los artículos encontrados que requieren conformidad y comparar con el número de artículos en la especificación que requieren conformidad. La fórmula es, una vez más, $X = 1 - A / B$ siendo A el número de artículos que no han sido implementados según los estándares de portabilidad pero que estaban especificados en la descripción del producto, el manual del usuario o el informe de especificaciones de pruebas. B sería el número total de artículos sobre cumplimiento de portabilidad que fueron especificados.

Por ejemplo, según la LOPD, para el tratamiento de datos de carácter personal de nivel alto es obligatorio que exista un registro de accesos a los datos. Si un programa que gestiona los historiales de los pacientes de un hospital no pide identificación a la hora de acceder a los historiales médicos

(datos de nivel alto) quiere decir que no está llevando un registro de acceso, por lo que este artículo se sumaría al conteo de A, restando así valor a la X e indicando que el software no es tan conforme con el cumplimiento de la portabilidad.

3.3 Métricas internas de portabilidad definidas en la ISO 9126-3 ³

Las métricas de portabilidad interna se usan para predecir el efecto que el producto software podría tener sobre el comportamiento del implementador o del sistema durante las actividades de portabilidad

Recordemos que las métricas internas deben ser utilizadas en un producto software no ejecutable durante sus etapas de desarrollo. Proporcionan a los usuarios la capacidad de medir la calidad de las entregas intermedias y por lo tanto predecir la calidad del producto final. Esto permite al usuario identificar los problemas de calidad e iniciar acciones correctivas lo antes posible en el ciclo de vida del desarrollo.

3.3.1 Métricas internas de adaptabilidad

Estas métricas indican un conjunto de atributos para predecir el impacto que el producto software puede tener sobre el esfuerzo del usuario que está intentando adaptar el producto software a diferentes entornos específicos.

El nombre de las métricas internas es el mismo que para las métricas externas, aunque para cada métrica cambia tanto la fórmula, los datos y a quién va dirigido. Los datos deben tomarse de la especificación de requisitos, del diseño y del informe de revisión. Estas métricas afectan a Desarrolladores, mantenedores y requeridores. Todas estas métricas deben ser confirmadas mediante sucesivas revisiones.

3.3.1.1 Adaptabilidad de las estructuras de datos

Esta métrica mide cuan adaptable es el producto a los cambios en las estructuras de datos. El método a aplicar para medirla es contar el número de estructuras de datos que son operables y que no tienen ninguna limitación después de la adaptación y compararlos con el número total de estructuras de datos que podrían requerir adaptaciones. La fórmula es $X = A / B$, siendo A el número de estructuras de datos que son operables y que no tienen ninguna limitación después de la adaptación, confirmadas en el informe de revisión y B el número total de estructuras de datos que tiene capacidad de adaptación. Cuanto más alto sea el valor de X, más adaptable es el producto software a las estructuras de datos.

Si volvemos al ejemplo del programa que contiene una base de datos de usuario compuesto por tuplas formadas por email, una contraseña y una foto. Si al portarlo a otro entorno programa es capaz de leer esos datos pero no es capaz de modificarlos, entonces esta estructura de datos se sumaría al conteo de B, pero no al de A.

3.3.1.2 Adaptabilidad al entorno de hardware

Esta métrica mide como de adaptable es el producto a un cambio en el entorno hardware, la medición se realiza mediante conteo. La fórmula a aplicar es la siguiente: $X = A / B$, siendo A el número de

³ Más concretamente en el apartado 6.8

funciones operativas cuyas tareas han sido completadas de manera totalmente satisfactoria durante el testeado de software en múltiples entornos hardware previamente especificados en el informe de revisión. La B es el número de funciones que se han examinado y que tenían capacidad de adaptación ante cambios del entorno hardware. Cuanto mayor sea el valor de la X, mejor.

En el ejemplo del programa que cuenta con el lector de huellas, si probamos 8 componentes hardware lectores de huellas y el programa falla con uno de ellos, la función “leer huella” incrementaría el contador de B, pero no el de A, ya que la tarea no ha sido completada de manera totalmente satisfactoria.

3.3.1.3 Adaptabilidad al entorno organizacional

Esta métrica mide cuan adaptable es el software a un cambio organizacional. En este caso la fórmula vuelve a ser $X = A / B$, siendo A el número de funciones implementadas que son capaces de adaptarse a múltiples organizaciones y entornos de negocio. B es el total de funciones relacionadas con cambios organizacionales. Cuanto mayor sea el valor de X, mejor.

Por ejemplo, si en una empresa se aumenta el número de empleados y estos deben estar conectados a un mismo servidor, si el servidor se satura debido al aumento de empleados, la función “conectar empleado” no aumenta el contador de A, pero si el de B.

3.3.1.4 Amigabilidad con el usuario para la portabilidad

Esta métrica mide la facilidad con se realizan actividades de portabilidad sobre el producto software. La fórmula a emplear es $X = A / B$ siendo B el número total de funciones que tienen posibilidades de adaptación fácil y A el número de funciones que permiten esa adaptación según el informe de revisión. Un valor de X más cercano a 1 indica mayor amigabilidad.

Por ejemplo, si en un programa se ha establecido que se deben portar y adaptar 7 funciones con relativa facilidad (la definición de “relativa facilidad” debe estar especificada en la especificación de requisitos), si tan sólo se han adaptado fácilmente 6, el valor de X sería 6/7

3.3.1.5 Adaptabilidad al entorno del sistema software

En esta métrica se mide la adaptabilidad del producto a los cambios relacionados con el entorno del sistema software. La fórmula vuelve a ser $X = A / B$, siendo A el número de funciones relacionadas con esta métrica que se adaptan a múltiples sistemas software y B el número total de funciones evaluadas. Cuanto más cercano a 1 sea el valor de X, más adaptable al sistema software será.

Por ejemplo: En una empresa hay un programa de contabilidad que conecta varios equipos mediante cable ethernet a un servidor central que funciona con Apache, si la función “conectar al servidor” no funciona después de cambiar el sistema operativo del servidor por uno Windows, esta función se sumaría al conteo de B pero no al de A, ya que no se adapta.

3.3.2 Métricas internas de instalabilidad

Estas métricas indican un conjunto de atributos para predecir el impacto que el producto software puede tener sobre el esfuerzo del usuario que está intentando instalar el producto software en un entorno de usuario específico. Estas métricas están orientadas a los desarrolladores de software. En

este caso se han añadido dos nuevas métricas que no estaban entre las externas: El esfuerzo en la instalación y la flexibilidad de la instalación.

3.3.2.1 Facilidad para reintentar la configuración

El propósito de esta métrica es conocer la facilidad con la que se puede repetir el proceso de configuración del software. La fórmula a aplicar es la siguiente: $X = A / B$ siendo A el número de operaciones implementadas para reintentar la configuración y B el número total de operaciones de configuración requeridas. Cuanto más cercano sea este valor a 1, más fácil de reconfigurar. Los datos deben obtenerse del informe de revisión.

Por ejemplo, si para configurar la conexión a un servidor, según el informe de revisión hacen falta 8 operaciones, pero una de ellas se ha repetido 3 veces, el valor de X sería 3/8

3.3.2.2 Esfuerzo en la instalación

Esta métrica mide el nivel de esfuerzo requerido para la instalación. Hay que contar el número de pasos de instalación automatizados implementados y compararlos con el número de pasos de instalación necesarios para alcanzar el objetivo. Los datos deben obtenerse del informe de revisión. La fórmula a aplicar es $X = A / B$, siendo A el número de pasos automatizados y B el número de pasos requeridos, Un mayor valor de X más cercano a 1 es mejor.

Por ejemplo, si en el informe de revisión de un programa pone que el número de pasos requeridos para una instalación exitosa es de 5, pero 2 de ellos no están automatizados (requieren interacción humana) el valor de X sería de 3/5, ya que 2 de los 5 pasos no están automatizados.

3.3.2.3 Flexibilidad en la instalación

Esta métrica no aparecía entre las métricas externas vistas anteriormente, e intenta medir cómo de flexible y personalizable es la instalación. Para la evaluación de esta métrica nos valdremos de la especificación de requisitos y del informe de revisión. Aplicaremos la fórmula $X=A/B$, siendo A el número de operaciones de instalación personalizables y B el número de operaciones de instalación personalizables requeridas. Un valor más cercano a 1 indica mayor flexibilidad.

Por ejemplo, si en el informe de revisión de un programa pone que el número de pasos requeridos para una instalación exitosa es de 5, pero 2 de ellos no están automatizados (requieren interacción humana) el valor de X sería de 3/5, ya que 2 de los 5 pasos no están automatizados.

3.3.3 Métricas internas de coexistencia

Dentro de este grupo tan sólo se encuentra una métrica, la cual nos indica un conjunto de atributos para predecir el impacto que el producto software puede tener sobre otro software compartiendo los mismos recursos hardware. Los datos se sacan de la especificación de requisitos, el informe de revisión y el informe de pruebas, los resultados le interesan a los requeridores, desarrolladores y mantenedores.

3.3.3.1 Posibilidad de coexistencia

Esta métrica mide cuan flexible es el producto compartiendo entorno con otros productos sin que haya impacto negativo sobre otros productos. Se calcula mediante la fórmula $X = A / B$, siendo A el

número de entidades con las cuales el producto puede coexistir y B el número de entidades en el entorno del producto que requieren coexistencia. Lo mejor es un valor de X lo más cercano a 1.

Por ejemplo, si en un programa de edición de textos el módulo de impresión (un módulo importante) no puede coexistir con el módulo de impresión del sistema operativo la entidad de módulo de impresión se sumaría al conteo de B pero no al de A.

3.3.4 Métricas internas de reemplazabilidad,

Estas métricas indican un conjunto de atributos para predecir el impacto del producto software tiene sobre el esfuerzo del usuario que está intentando usar el software en lugar de otro software específico en un entorno y contexto de uso específicos. Los datos se toman de Del diseño, el código fuente, el informe de revisión y el informe de pruebas, los resultados le interesan a los requeridores, desarrolladores y mantenedores.

3.3.4.1 Uso continuado de los datos

Esta métrica mide la cantidad de datos originales que permanecen intactos después del reemplazamiento del producto. La fórmula a aplicar vuelve a ser $X = A / B$, siendo A el número de datos que continúan usándose después del reemplazamiento y B el número de datos que había que reutilizar del antiguo software. Un valor de X más cercano a 1 es lo mejor.

Si ponemos como ejemplo un programa de gestión de nóminas que es reemplazado por otro programa de similares características que debe reutilizar los siguientes datos: Información de los empleados, historial de nóminas, nómina en curso y vida laboral. Si conseguimos reutilizar todos los datos excepto el historial de nóminas debido a un conflicto de formatos, el valor de X sería 3/4

3.3.4.2 Inclusividad de funciones

Esta métrica mide la cantidad de funciones que permanecen intactas después del cambio. La fórmula a aplicar es $X = A / B$, siendo A el número de funciones reemplazadas por el nuevo software que producen resultados similares a las funciones del software antiguo. B es el número de funciones en el viejo software. Cuanto más cercano a 1 sea el valor de X, el cumplimiento de esta métrica es mayor.

Por ejemplo, al sustituir el software de gestión de nóminas (que tenía 14 funciones) por otro de características similares se ha descubierto que la función “descargar nómina” no existe, por tanto el valor de X es de 13/14

3.3.5 Métricas internas de cumplimiento de la portabilidad

Estas métricas indican un conjunto de atributos para evaluar la capacidad del producto software para cumplir con los estándares, convenciones o regulaciones que tienen que ver con la portabilidad. Tan sólo encontramos una métrica, la cual se nutre de la especificación de cumplimiento de estándares, el diseño, el código fuente y el informe e revisión. Interesa a los requeridores y a los diseñadores.

3.3.5.1 Cumplimiento de la portabilidad

Esta métrica mide cómo de conforme es la portabilidad del producto con regulaciones, estándares y convenciones aplicables. El método a aplicar es contar los artículos encontrados que requieren conformidad y comparar con el número de artículos en la especificación que requieren conformidad.

La fórmula es $X = A / B$ siendo A el número de artículos que han sido implementados según los estándares de portabilidad y B es el número total de artículos sobre el cumplimiento de portabilidad.

Por ejemplo, el programa que gestiona los historiales de los pacientes de un hospital está formado por 20 módulos, de los cuales 14 cumplen con la LOPD al implementar un registro de acceso a los datos de carácter personal de nivel alto (los historiales médicos). Los 6 módulos restantes no implementan el registro de acceso, por lo cual el valor de X sería $1 - 6/20$

3.4 Nuevas métricas de portabilidad

Como complemento a las métricas que vienen en la ISO/IEC 9126, se proponen siete métricas nuevas, de las cuales seis son referidas a la adaptabilidad y una a la instalabilidad. De ellas, tres son métricas internas y cuatro son métricas externas. Estas nuevas métricas son las siguientes:

- Métricas externas:
 - Adaptabilidad a distintos dispositivos
 - Adaptabilidad al servidor de aplicaciones
 - Instalabilidad frente al reemplazo del sistema operativo
- Métricas internas:
 - Adaptabilidad ante el cambio de *framework* de desarrollo
 - Adaptabilidad del código: Autocontención o encapsulamiento de módulos
 - Adaptabilidad del código: Autodocumentación de formas estructurales
 - Adaptabilidad del código: Especificación de módulos

3.4.1 Adaptabilidad a distintos dispositivos

Esta métrica trata de cuantificar el grado de adaptabilidad de un producto software a distintos dispositivos tales como ordenadores portátiles, tablets, smartphones, e-readers, smartTVs, smartwatches, etc. Es una métrica muy actual, ya que hoy en día la adaptabilidad de un software es muy importante para poder llegar al mayor número de usuarios posibles.

Se calcula mediante la fórmula $X = 1 - A / B$, siendo B el número de dispositivos especificados en los que el producto software debe ser adaptable y A el número de dispositivos en los que la adaptabilidad no es del todo satisfactoria. Un valor de X cercano a 1 es lo más deseable.

Por ejemplo, si en una aplicación web en la que en la especificación de requisitos dice que debe ser adaptable a 4 dispositivos (portátiles, tablets, smartphones, e-readers), pero en las pruebas con el usuario la adaptabilidad al e-reader no es la adecuada (letras demasiado grandes y secciones que se tapen unas a otras) entonces el valor de X sería igual a $1 - 1/4$

3.4.2 Adaptabilidad al servidor de aplicaciones

Esta métrica mide el grado de adaptabilidad de un producto software ante un cambio en el servidor de aplicaciones del que depende. Hoy en día multitud de software depende de un servidor de aplicaciones, sobre todo muchas páginas web, las cuales ejecutan aplicaciones en el servidor de aplicaciones y muestran su contenido a los usuarios.

Mediante esta métrica se persigue conocer cuan dependiente es un producto software del servidor comercial. Teniendo en cuenta solamente la dimensión económica o comercial, es muy interesante

que un producto software pueda adaptarse o portarse a distintos servidores de aplicaciones. De esta forma de persiguen, entre otros, los siguientes objetivos:

1. Reducir los costes de adaptabilidad o portabilidad de un producto software a distintos servidores de aplicaciones.
2. Incrementar el número de posibles clientes
3. Aumentar la calidad del código fuente del producto al no tener que hacer distintas adaptaciones, métodos, funciones, etc. según el servidor de aplicaciones correspondiente.

Se calcula mediante la fórmula $X = A / B$, siendo B el número de aplicaciones especificadas que deben ser adaptadas al nuevo servidor de aplicaciones y A el número de aplicaciones a las que el producto se adapta perfectamente después del cambio. El valor de X debe oscilar entre 0 y 1, siendo el 1 el valor más deseable, pues significa que todas las aplicaciones del servidor han sido correctamente adaptadas.

Por ejemplo, si nuestro servidor de aplicaciones utiliza un software de IBM WebSphere podemos querer cambiarlo por cuestiones estratégicas por el software BEA WebLogic (que es un software que se adapta muy bien a los distintos clientes, independientemente del sistema operativo que utilicen). Este servidor de aplicaciones ejecuta 30 aplicaciones de nuestro producto software, de las cuales todas funcionan correctamente después de la sustitución del software del servidor de aplicaciones, por lo que el valor de X sería uno. Esto quiere decir que, desde el punto de vista de esta métrica, las aplicaciones o productos software tienen un nivel alto de adaptabilidad alto (en realidad el máximo que es uno) al cambio del servidor de aplicaciones.

3.4.3 Instalabilidad frente al reemplazo del sistema operativo

Esta métrica intenta medir el grado de instalabilidad del producto software frente a un cambio del sistema operativo en el que se ejecuta. Se diferencia de la adaptabilidad al entorno software en que esta métrica contempla los efectos de un cambio total del entorno en el que se ejecuta (el Sistema Operativo) y la métrica de adaptabilidad está más bien referida a cambios en los componentes software del entorno.

Se calcula mediante la fórmula $X = A / B$, siendo B el número de módulos que según la especificación de requisitos deben ser instalables en el nuevo sistema operativo y A el número de módulos que se han instalado de manera satisfactoria. El valor de X debe estar 0 y 1, siendo el 1 el valor más deseable, pues significa que todos los módulos del producto software han sido correctamente instalados.

Por ejemplo, un programa de gestión de una empresa de logística está formado por 15 módulos, el programa se ha estado ejecutando en un ordenador con Windows GNU/Linux). Al intentar instalar el programa en el nuevo sistema operativo se ha conseguido instalar satisfactoriamente todos los módulos excepto el módulo de geolocalización de camiones por GPS, que si bien se ha conseguido instalar, muestra los camiones con mucho retraso. El valor de X sería 14/15, ya que la instalación del módulo de geolocalización no es del todo satisfactoria debido a su lentitud.

3.4.4 Adaptabilidad ante el cambio del *framework* de desarrollo

Esta métrica mide el impacto que tiene el cambio del *framework* de desarrollo sobre la portabilidad del producto software. En muchas ocasiones el *framework* en el que se está desarrollando el producto no convence en cuanto a prestaciones y funcionalidad y se hace necesario cambiarlo por otro más potente. Es una métrica interna, ya que se refiere a las fases de desarrollo del producto software

Se calcula mediante la fórmula $X = A / B$, siendo B el número de funciones que funcionaban con el *framework* anterior y A el número de funciones que funcionan con el nuevo *framework* de manera satisfactoria. El valor de X debe estar 0 y 1, siendo el 1 el valor más deseable, pues significa que todas las funciones del producto funcionan en el nuevo *framework*.

Por ejemplo, en el desarrollo de una aplicación web se ha utilizado el *framework* web Fundación, pero llegados a un punto, se decide cambiar Foundation por Bootstrap, ya que Bootstrap es totalmente compatible con el preprocesador de CSS LESS. En el proceso de cambio de *framework* se han trasladado las 106 funciones de Foundation a Bootstrap, pero una de ellas, la función “menú desplegable” no funciona correctamente, ya que no despliega el menú si no le añades código JavaScript. Por tanto el valor de X sería de 105/106, ya que la función “menú desplegable” requiere de un fichero extra para funcionar satisfactoriamente.

3.4.5 Métricas para la adaptabilidad del código

Estas métricas contribuyen a la portabilidad del producto software creando un código más legible y más inteligible, y por lo tanto más fácil de transferir de un entorno a otro. Todas son de tipo interno, ya que se refieren exclusivamente a cómo está hecho el sistema y afectan a las fases de desarrollo del producto. Todas estas métricas están orientadas a desarrolladores.

3.4.5.1 Autocontención o encapsulamiento de módulos

Esta métrica mide el nivel de encapsulamiento del código, es decir, que las variables, las constantes y los tipos de datos sean utilizados sólo en el contexto en el que son definidos. Un código correctamente encapsulado evita que al transferir un módulo a otro entorno nos encontremos con referencias a variables y a tipos de datos que no existen en el nuevo entorno.

La fórmula que propongo aplicar es $X = R/N$, siendo R el número total de referencias no válidas en el nuevo entorno y N el número de módulos que hemos transferido a otro software. Un valor de X de 0 o lo más cercano posible indica que el código está perfectamente encapsulado

Por ejemplo, si un módulo “ordenar” de un software de gestión de bibliotecas utiliza el tipo de datos “info de libro” pero no lo declara dentro del módulo, sino que lo toma de otro del que no depende necesariamente (como por ejemplo, el módulo “opiniones”), entonces el tipo de datos “info de libro” se sumaría al conteo de N, pero no al de R, lo que reduciría el valor de X, lo que significa una menor calidad en el encapsulamiento.

3.4.5.2 Autodocumentación de formas estructurales

Esta métrica trata de medir el grado de documentación de una forma estructural (módulos, ciclos, estructuras de datos, variables, constantes, tipos, etc.). Una forma estructural está documentada cuando su propósito y propiedades están explícitamente definidos en el contexto de la forma estructural. Un adecuado uso de los nombres de funciones y variables también ayudan a la

autodocumentación. Cuanto mejor documentadas estén las formas estructurales en el código, más sencillo resulta para un desarrollador el transferirla a otro entorno o simplemente reutilizarla dentro del mismo programa.

Se calcula mediante la fórmula $X = 1 - A / B$, siendo B el número total de formas estructurales y A el número de formas estructurales que no están correctamente documentadas. El valor de X estará entre 1 y 0, siendo 1 o el valor más cercano a 1 lo que indica que el producto software está correctamente documentado.

Si seguimos con el ejemplo del software de gestión de bibliotecas, el tipo de datos “info de libro” no tiene documentado cómo rellenar sus campos, aparece como un struct con distintos datos pero sin nombres explicativos ni ningún tipo de comentario. Por tanto el tipo de datos “info de libro” se sumaría tanto al conteo de A como al de B, resultando perjudicial, ya que aleja el valor de X del 1.

3.4.5.3 Especificación de módulos

Esta métrica trata de medir cuan especificados están los módulos de un producto software. Un módulo está especificado si su funcionalidad es descrita con precondiciones y postcondiciones. Un módulo correctamente especificado ahorra mucho tiempo a los desarrolladores, ya que evitan tener que revisar el código para “adivinar qué hace” el módulo.

Se calcula mediante la fórmula $X = 1 - A / B$, siendo B el número total de módulos y A el número de módulos que no están correctamente especificados o que directamente carecen de precondiciones o postcondiciones. El valor de X estará entre 1 y 0, siendo 1 o el valor más cercano lo que indica que el producto software está correctamente documentado.

Por ejemplo, el programa de gestión de una biblioteca tiene 13 módulos, de esos 13, a primera vista hay 2 que no se sabe exactamente qué hacen ya que no tienen ni precondiciones ni postcondiciones definidas. Después de examinar detenidamente el contenido de los módulos descubrimos que son para exportar la ficha de un libro y para el sistema de reservas. Por tanto el valor de X sería $1 - 2/13$.

3.5 Tabla resumen de las métricas

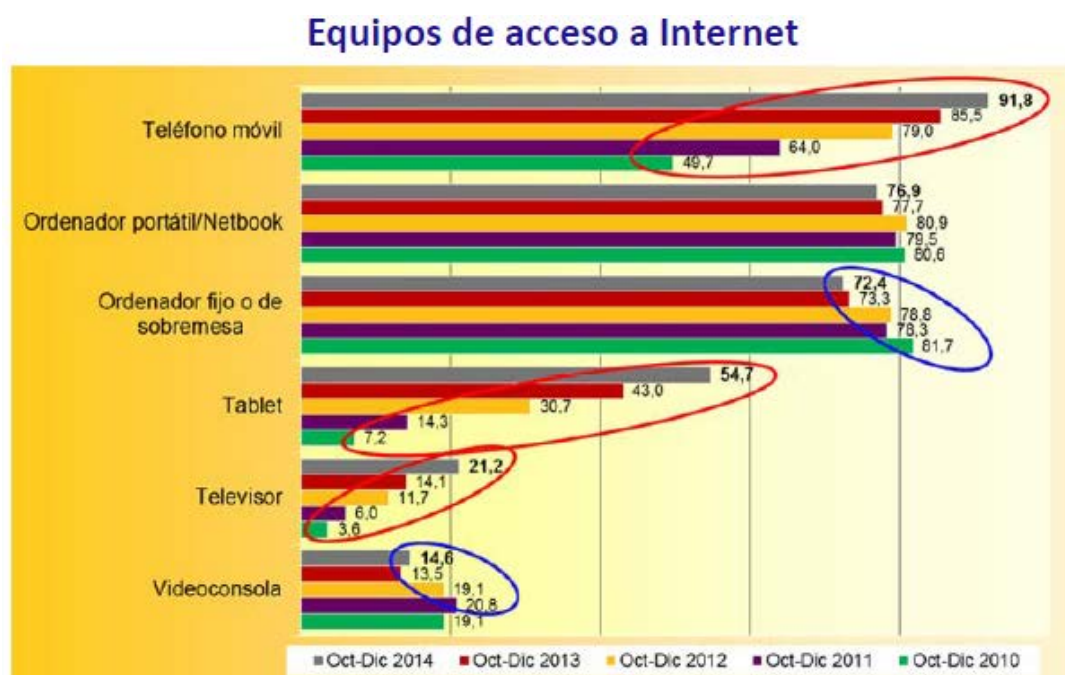
A continuación se muestra una tabla que resume todas las métricas expuestas en este capítulo, con la finalidad de poder verlas todas en su conjunto. La tabla se puede ver en el anexo A.

4. Caso práctico

4.1. Introducción

Con el fin de poner en práctica las métricas expuestas en el capítulo anterior, se realizará un caso práctico en el cual se evaluará la calidad de la portabilidad en distintas aplicaciones. Las aplicaciones elegidas son cuatro páginas web de distintas universidades españolas (UCM, UAM, UOC y UC3M)

La portabilidad es un factor muy importante si hablamos de aplicaciones web, sobre todo en cuanto a la adaptabilidad, ya que interesa que la aplicación web esté disponible en el mayor número de dispositivos y navegadores para así poder llegar a un mayor número de gente. Es importante tener presente que -cada vez más- la gente se conecta a Internet con diferentes dispositivos.



Encuesta de la AIMC⁴ que muestra en percentiles los equipos de acceso a Internet y su evolución

4.2 Descripción del laboratorio de auditoría y aplicaciones a ser auditadas

4.2.1 Laboratorio de auditoría

Para la realización del caso práctico se cuenta con las siguientes herramientas hardware:

- Ordenador portátil Acer Travel Mate 5742Z, con procesador Intel Pentium P6100 a 2 GHz, 6 GB de memoria RAM, 500GB de disco duro y pantalla de 17,5 pulgadas. Tiene Windows 8.1 y Ubuntu 14.04
- Ordenador MacBook Pro con procesador Intel Core i5 a 2.7 GHz, 8GB de memoria RAM y 128 GB de disco duro SSD con sistema operativo OS X Yosemite.
- Tablet Samsung Galaxy Tab 4, con CPU Quad-Core a 1.2 GHz, 1.5 GB de memoria RAM y 16 GB de memoria interna, pantalla de 10,1 pulgadas y Android 4.2.2.

⁴Asociación para la investigación de medios de comunicación

- E-Reader Kindle 4, con pantalla de 4 pulgadas y sistema operativo Duokan.
- Smartphone Samsung Galaxy S7580 con chipset Dual Core a 1.2 GHz, 768 MB de memoria RAM, 1.79 GB de memoria interna y pantalla de 4 pulgadas.

No se cuenta con un smartwatch debido a su alto coste (100 € el modelo más económico), pero aun así se incluye como futura línea de investigación.

También se dispone del siguiente software instalado en algunos dispositivos

- En el ordenador están instalados los siguientes navegadores: Chrome 44, Firefox 33 e Internet Explorer 11 en Windows y los navegadores Chrome 45, Firefox 40 y Opera 33 en Ubuntu.
- En el Mac están instalados los navegadores Chrome 42, Firefox 40, Opera 31 y Safari 8.
- En la Tablet están instalados los navegadores Chrome 44, Firefox 40 y Opera 10.
- El smartphone tiene instaladas las aplicaciones de Chrome 44, Firefox 40 y Opera 10.
- El e-reader viene con un navegador integrado en Duokan.

Lo cual suma un total de 5 dispositivos, 5 sistemas operativos y 6 navegadores.

4.2.2 Aplicaciones a ser auditadas

Para la realización de este caso práctico se auditarán las siguientes aplicaciones web:

- Página web de la Universidad Complutense de Madrid (UCM): www.ucm.es
- Página web de la Universidad Carlos III de Madrid (UC3M): www.uc3m.es
- Página web de la Universidad Autónoma de Madrid (UAM): www.uam.es
- Página web de la Universitat Oberta de Catalunya (UOC): www.uoc.edu

Se han elegido porque ofrecen el mismo tipo de servicios y soportan una gran cantidad de usuarios al día.

4.3 Objetivos

Se parte de la premisa de que las aplicaciones a auditar fueron inicialmente concebidas para su uso en un navegador web, concretamente en un PC Windows con Internet Explorer. Con el paso del tiempo las webs se han ido actualizando tanto en contenido como en diseño, pero casi siempre orientado a ordenadores, no a otro tipo de dispositivos.

Los objetivos de este caso práctico son los siguientes:

- Comprobar la adaptabilidad de las cuatro aplicaciones a distintos dispositivos.
- Establecer una clasificación de las aplicaciones en base a la portabilidad.
- Proponer mejoras en la adaptabilidad de las aplicaciones.

4.4 Método (actividades del 25040)

Para evaluar la adaptabilidad de las aplicaciones se utilizará el referente ISO/IEC 25040. Más concretamente el capítulo 6, que detalla el proceso de evaluación de la calidad del producto.

El referente especifica en el punto 6.1 -requisitos generales- que el personal implicado debe tener el entrenamiento y recursos necesarios. Los directores de este TFG han avalado el cumplimiento del punto anterior por parte del alumno autor de este trabajo.

4.4.1 Establecer los requisitos de la evaluación

Esta primera actividad consta de cuatro tareas, la primera de ellas es establecer el propósito de la evaluación, el cual es comparar los productos entre si y evaluar los aspectos positivos y negativos mientras este se utiliza.

Dado que el producto está en su madurez, el tipo de evaluación que se hará será de tipo externo. Se evaluarán las siguientes partes del producto:

- La web de portada
 - UCM: www.ucm.es
 - UC3M: www.uc3m.es
 - UAM: www.uam.es
 - UOC: www.uoc.edu
- La página web de contacto
 - UCM: www.ucm.es/contacto
 - UC3M: www.uc3m.es/contacta
 - UAM: www.uam.es/contacto
 - UOC: <http://www.uoc.edu/portal/es/universitat/contacte-seus/>
- El directorio de personal
 - UCM: fdi.ucm.es/Directorio⁵
 - UC3M: aplicaciones.uc3m.es/directorio
 - UAM: autoservicio.uam.es/paginas-blancas
 - UOC: www.uoc.edu/opencms_portal2/opencms/ES/universitat/contacte-seus/directori-persones/list.html

El referente nos indica que el rigor de la evaluación debe definirse en función del propósito y uso previsto del producto software. Puesto que las cuatro aplicaciones web de este caso no tienen riesgos para la seguridad, económicos o ambientales, el rigor de la evaluación será medio.

4.4.2 Especificar la evaluación

Consta de tres tareas, la primera de ellas es la selección de métricas de calidad. Es aquí donde se seleccionan las métricas de calidad que deben cubrir los requisitos de evaluación. Se han seleccionado las siguientes:

- Adaptabilidad al entorno del sistema software
- Adaptabilidad a distintos dispositivos

Puesto que tan sólo se tiene acceso a la interfaz web de la aplicación, el resto de métricas vistas en el capítulo anterior no son aplicables a este caso práctico.

⁵ Se evaluará el directorio de la web de la Facultad de Informática debido a que la página de la UCM no cuenta con un directorio.

Para la adaptabilidad al entorno del sistema hardware tendremos en cuenta tanto el sistema operativo, como el navegador, por lo que la métrica “Adaptabilidad al entorno del sistema software” se desdobra en 2 submétricas, por lo que la selección de métricas quedaría de la siguiente forma:

- Adaptabilidad al entorno del sistema software: Sistema Operativo
- Adaptabilidad al entorno del sistema software: Navegador
- Adaptabilidad a distintos dispositivos

La siguiente tarea es definir los criterios de decisión para las métricas, por lo cual los umbrales numéricos para las métricas seleccionadas son los siguientes:

Escala de medición	Niveles de puntuación	Grado de satisfacción
0 - 0.3	Inaceptable	Insatisfactorio
0.4 – 0.6	Aceptable	Satisfactorio
0.7 - 1	Correcto	Muy satisfactorio

Por último, la tercera tarea es definir los criterios de decisión para la evaluación, que son los siguientes:

- El tamaño de las letras hace posible su lectura
- Los elementos de la página no se superponen entre ellos
- Los encabezados y pies de página están en el lugar correcto
- Se muestran todos los elementos
- Los menús desplegables funcionan
- Las animaciones, imágenes y transiciones se visualizan correctamente

4.4.3 Diseñar la evaluación

Esta actividad tan solo tiene una tarea, que es planificar las actividades de la evaluación.

A la hora de planificar la evaluación de las 4 aplicaciones web en los distintos dispositivos y con distintos sistemas nos damos cuenta de que hay múltiples combinaciones de dispositivo-sistema operativo-navegador, por lo que se decide crear los siguientes perfiles:

- Ordenador Windows con navegador Chrome
- Ordenador Windows con navegador Opera
- Ordenador Windows con navegador Internet Explorer
- Ordenador Windows con navegador Firefox
- Ordenador Linux con navegador Chrome
- Ordenador Linux con navegador Opera
- Ordenador Linux con navegador Firefox
- Ordenador Mac con navegador Chrome
- Ordenador Mac con navegador Opera
- Ordenador Mac con navegador Firefox
- Ordenador Mac con navegador Safari
- Móvil Android con navegador Chrome

- Móvil Android con navegador Opera
- Móvil Android con navegador Firefox
- Tablet Android con navegador Chrome
- Tablet Android con navegador Opera
- Tablet Android con navegador Firefox
- E-reader Kindle con sistema operativo Duokan y navegador nativo.

Con lo que el número de perfiles a evaluar se queda en 18.

Cabe reseñar que Internet Explorer tan solo funciona en plataformas Windows, por lo que no ha podido ser evaluado en el resto de sistemas operativos. Lo mismo pasa con el navegador Por otro lado indicar que aunque el e-reader Kindle de Amazon trae su propio sistema operativo, se ha decidido instalar el sistema operativo Duokan, el cual además de permitir leer en múltiples formatos, incluye un navegador mejor que el que trae el dispositivo en su sistema operativo por defecto.

Respecto a la planificación temporal se han realizado todas las mediciones en la misma tarde.

Se han creado unas *checklist* en modo de tablas para poder realizar un seguimiento de los aspectos medidos, se ha creado una por aplicación web. Se puede ver en el Anexo B.

4.4.4 Ejecutar la evaluación

Esta etapa se compone de 3 pasos, el primero es realizar las mediciones, para ello se han utilizado los dispositivos y se ha navegado con ellos por los 4 productos software para así rellenar las tablas de las *checklist*. Igualmente se han apuntado en los comentarios los motivos que hacen que no se consiga satisfacer algún criterio en concreto. El resultado se puede ver en los anexos C (UCM), D (UAM), E (UC3M) y F (UOC), para una mejor visualización se ha utilizado un código de colores: verde si cumple el criterio, rojo si no lo cumple y gris si no es aplicable.

Durante la evaluación se ha descubierto que la página de la UCM ha decidido eliminar el directorio, un repaso a internet.archive.org⁶ nos demuestra que el directorio desapareció en febrero de 2013 con la remodelación de la web. Aunque si escribes el nombre de un profesor en el buscador, aparecen sus datos de contacto. Es por esto que se evaluará el directorio de la Facultad de Informática. También se ha descubierto que el buscador de la UOC no funciona en su versión móvil.

Imagen de la web de la UCM con el directorio

Un dato a tener en cuenta son las redirecciones que efectúan algunas aplicaciones sirviéndote una web u otra dependiendo del dispositivo desde el que se navegue. El mejor ejemplo es el de la UOC, ya que si pones la dirección www.uoc.edu en un smartphone, la aplicación te redirige a la dirección www.uoc.mobi/opencms_portal2/opencms/mobi/es/index.html que cambia incluso del dominio .edu al .mobi.

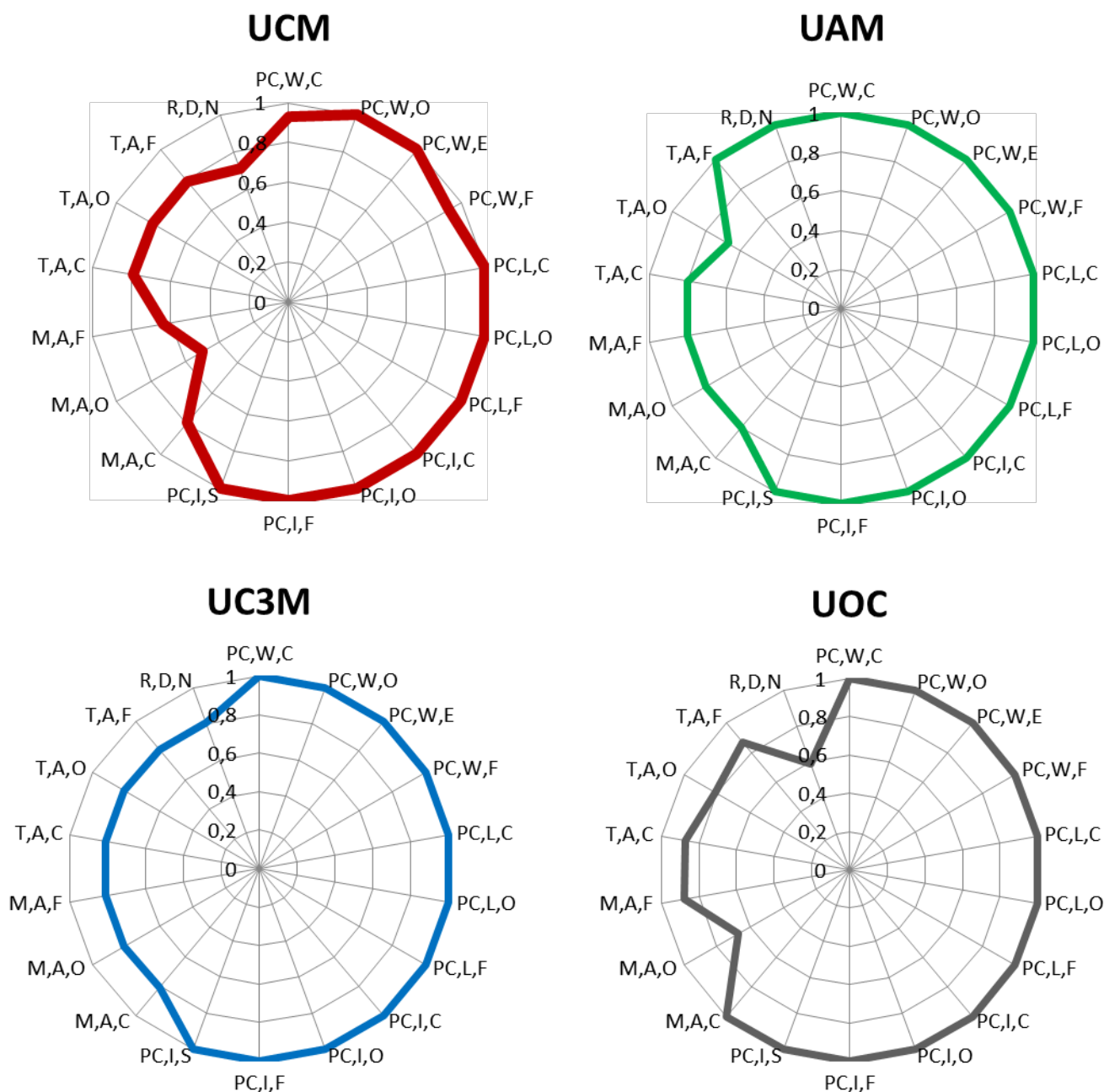
Después se ha calculado el valor de X de cada uno de los perfiles, dividiendo el número de mediciones positivas entre el número de mediciones realizadas. El valor de X se ha aproximado a la centésima.

⁶ Internet Archive es una asociación sin ánimo de lucro que guarda un historial de Internet.

4.4.5 Concluir la evaluación:

Una vez efectuadas las mediciones se han repasado las mediciones con el fin de detectar errores, se han detectado y corregido 2 errores fruto de descuidos (un ✓ que debería estar una columna a la derecha y un N/A de sobra por culpa del copia-pegar).

Después se han puesto los datos en forma de gráfico, para que así sea más sencilla su interpretación.



Leyenda

Dispositivos: **PC**, Móvil Smartphone, Tablet, e-Reader
 Sistema Operativo: **Windows**, Linux, **IOS**, Android, Duokan
 Navegador: **Chrome**, Opera, Firefox, Internet Explorer, Safari, Nativo Duokan
 Páginas: **Principal**, Contacto, Directorio

Gráficos de resultados

4.5 Conclusiones

Como podemos ver, las aplicaciones analizadas se encuentran dentro del rango que hemos definido como correcto, excepto en el caso de la UCM visualizada desde un smartphone Android con Opera y el caso de la web de la UOC visualizada desde un e-Reader que se encuentran en el rango que hemos definido como aceptable.

También llama la atención la diferencia mostrada entre navegadores en un mismo sistema operativo, es destacable la diferencia en cuanto a posicionamiento y tamaño de elementos se refiere.

La aplicación web de la UCM no está preparada para la adaptabilidad a dispositivos móviles pese a su reciente cambio de imagen. Un fallo grave que presenta es que al ampliar, el pie de página se coloca en el medio de la pantalla y no permite ver el contenido que hay detrás.

La aplicación web de la UAM funciona correctamente en todos los dispositivos, aunque el tamaño excesivamente pequeño de la letra obliga a hacer zoom en los smartphone.

Sin embargo la aplicación web de la UC3M es la que mejor adaptabilidad presenta, esto es gracias a su “css responsive”, lo que le permite adaptarse al dispositivo. El css responsive es una filosofía de diseño y desarrollo cuyo objetivo es adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visualizarla.

La aplicación web de la UOC, aunque se visualiza correctamente en todos los dispositivos tiene una peculiaridad: oculta información al usuario de smartphone mostrándole una versión reducida de su página web. Además de que hay secciones de la web móvil que no funcionan correctamente (como el buscador interno).

Por tanto, la aplicación web de la UC3M es la más portable de las analizadas, seguida de la de la UAM, la UOC y por último la UCM. Esto es debido al “css responsive” que tiene la web de la UC3M, la cual sigue las recomendaciones de la WC3 para el diseño adaptable a distintos dispositivos.

5. Conclusiones y líneas abiertas

5.1 Conclusiones

Tanto la ISO/IEC 9126 como la ISO/IEC 14598 evalúan, entre otras características, la portabilidad en la calidad del producto software, y aunque una complementa a la otra, se hace necesaria la familia ISO/IEC 25000 para hacer converger estos dos referentes.

Se han analizado e investigado las métricas propuestas en los estándares anteriores detectando debilidades en dichos estándares. Por ello en este TFG se proponen nuevas métricas que persiguen mejorar los resultados a obtener en las auditorías de calidad del producto software.

Las métricas mostradas en la ISO/IEC 9126 son insuficientes, y se hace necesaria la adición de nuevas métricas para conseguir un referente de calidad más exhaustivo y, por lo tanto, auditorías de calidad software más completas y que aporten mayor valor a las organizaciones. En este TFG se proponen nuevas métricas que complementan y enriquecen a las existentes en los estándares analizados e investigados.

Como se ha podido comprobar en el caso práctico, la adaptabilidad de la web de nuestra universidad es mejorable, siendo la opción de más calidad usar un diseño “css responsive” como hace la UC3M.

5.2 Líneas abiertas

Las principales líneas abiertas de investigación estarían en relación con la portabilidad del producto software:

1. A nuevos dispositivos como los smartwatches.
2. A distintas características los smartphones relacionados con el tamaño de la pantalla. En las recientes presentaciones de los productos más conocidos se comprueba una tendencia al aumento en el número de pulgadas de las pantallas. Este aumento físico del dispositivo favorece de una forma natural la portabilidad de aplicaciones diseñadas para dispositivos tradicionales como los PC. No obstante surgen nuevos retos como el aprovechar el uso incipiente de estos nuevos dispositivos desde un punto de oportunidad comercial.
3. Propuestas de dentro el ciclo de vida de desarrollo software orientadas a fomentar la portabilidad desde las etapas tempranas de dicho ciclo.
4. Profundizar en la investigación de las distintas subcaracterísticas de la portabilidad con el fin de definir nuevas métricas a tener en cuenta en el desarrollo de aplicaciones y en la auditoría de las mismas.
5. Analizar y proponer relaciones entre las características de accesibilidad y la adaptabilidad de aplicaciones al tener puntos en común como ha quedado reflejado en el caso práctico presentado. De esta forma se podrían llegar a proponer métodos de auditorías comunes para la accesibilidad y la adaptabilidad. Con ello se tendría una ventaja comercial en la ejecución de estas auditorías ofreciendo más valor a los potenciales clientes.

Anexo A: Tabla resumen de las métricas

		Nombre	¿Qué miden?	Fórmula	Valor de X
ISO/IEC 9126-2	EXTERNAS	Adaptabilidad			
		Adaptabilidad de estructuras de datos	Si el usuario o el encargado de mantenimiento pueden adaptar de manera sencilla el software a los nuevos tipos de datos en el nuevo entorno.	$X = A / B$ <p>A = Número de estructuras de datos que son operables. B = Número de datos que se espera que sean operables en el nuevo entorno.</p>	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Adaptabilidad al entorno Hardware	La facilidad con la que un usuario o un mantenedor adaptan el software al entorno. Cómo el software es capaz de adaptarse por sí mismo al nuevo entorno.	$X = 1 - A / B$ <p>A = Número de funciones operativas cuyas tareas no han sido completadas de manera totalmente satisfactoria durante el testeo de software en el entorno hardware. B = Número de funciones examinadas.</p>	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Adaptabilidad al entorno organizacional	Cómo de adaptable es el software a un cambio organizacional.	$X = 1 - A / B$ <p>A = Número de funciones implementadas que no son capaces de adaptarse a múltiples organizaciones. B = Número de funciones analizadas.</p>	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Amigabilidad para la portabilidad	El esfuerzo invertido por el usuario realizando operaciones de portabilidad sobre el producto.	$X = T$ <p>T = Suma del tiempo invertido en completar la adaptación del software al entorno del usuario cuando este intenta instalar o configurar el software.</p>	$0 \leq X$ Cuanto mayor sea X, mejor.
		Adaptabilidad al entorno del sistema software	La adaptabilidad del producto software al sistema operativo o software de red. Cuan adaptable es el software a los cambios en el entorno relacionados con el sistema software.	$X = A / B$ <p>A = Número de funciones relacionadas con esta métrica que se adaptan al sistema software. B = Número total de funciones evaluadas.</p>	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
NUEVAS		Adaptabilidad a distintos dispositivos	El grado de adaptabilidad de un producto software a distintos dispositivos tales como ordenadores portátiles, tablets, smartphones, e-readers, smartTVs, smartwatches...	$X = 1 - A / B$ <p>A = Número de dispositivos en los que la adaptabilidad no es del todo satisfactoria. B = Número de dispositivos especificados en los que el producto software debe ser adaptable.</p>	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.

		Nombre	¿Qué miden?	Fórmula	Valor de X
NUEVAS	EXTERNAS	Adaptabilidad al servidor de aplicaciones	El grado de adaptabilidad de un producto software ante un cambio en el servidor de aplicaciones del que depende.	$X = A / B$ A = Número de aplicaciones a las que el producto se adapta perfectamente después del cambio. B = Número de aplicaciones especificadas que deben ser adaptadas al nuevo servidor de aplicaciones.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Instalabilidad			
ISO/IEC 9126-2		Facilidad de Instalación	Cuan fácil es para un usuario instalar por primera vez el software en un entorno nuevo.	$X = A / B$ A = Número de veces que el usuario ha tenido éxito en cambiar la instalación. B = Número de veces que el usuario lo ha intentado, hayan tenido éxito o no.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Facilidad de Re-Instalación	Si el usuario es capaz de reinstalar y configurar el software.	$X = 1 - A / B$ A = Número de fallos del usuario al intentar reinstalar el software. B = Número de intentos.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
NUEVAS		Instalabilidad frente al reemplazo del sistema operativo	El grado de instalabilidad de producto software frente a un cambio del sistema operativo en el que se ejecuta.	$X = A / B$ A = Número de módulos que se han instalado de manera satisfactoria. B = Número de módulos que según la especificación de requisitos deben ser instalables en el nuevo sistema operativo.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Coexistencia			
ISO/IEC 9126-2		Posibilidad de coexistencia	Si es posible la coexistencia con otro software en el mismo entorno.	$X = A / T$ A = Número de restricciones o fallos inesperados que el usuario encuentra durante el uso del software de manera concurrente con otro software. T = Tiempo que se emplean de manera simultánea el software a evaluar y otros de frecuente uso.	$0 \leq X$ Cuanto más cercano a 0, mejor.
		Reemplazabilidad			
		Uso continuado de los datos	Si el usuario o el “mantenedor” son capaces de seguir usando los mismos datos que usaba con el software anterior después de haber sido reemplazado por uno nuevo.	$X = A / B$ A = Número de datos que se usaban en el software anterior y que se pueden seguir utilizando en el nuevo software. B = Número de datos que se usaban en el software anterior y de los cuales se planea su reutilización en el nuevo software.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.

		Nombre	¿Qué miden?	Fórmula	Valor de X
ISO/IEC 9126-2	EXTERNAS	Inclusividad de funciones	La facilidad con la que un usuario o mantenedor puede continuar utilizando funciones similares después de reemplazar el software.	$X = A / B$ <p>A = Número de funciones que producen resultados similares a los producidos por el software anterior y que no necesitan ser cambiadas.</p> <p>B = Número de funciones similares a las funciones del software a reemplazar.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>
		Consistencia funcional al soporte del usuario	Cómo de consistentes son los nuevos componentes con los ya existentes en la interfaz de usuario.	$X = 1 - A / B$ <p>A = Número de nuevas funciones que el usuario encuentra inaceptables e inconsistentes con sus expectativas.</p> <p>B = Número de nuevas funciones.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>
		Cumplimiento de la portabilidad			
		Cumplimiento de la portabilidad	Cómo de conforme es la portabilidad del producto con regulaciones, estándares y convenciones aplicables.	$X = 1 - A / B$ <p>A = Número de artículos que no han sido implementados según los estándares de portabilidad pero que estaban especificados en la descripción del producto, el manual del usuario o el informe de especificaciones de pruebas.</p> <p>B = Número total de artículos sobre cumplimiento de portabilidad que fueron especificados.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>
ISO/IEC 9126-3	INTERNAS	Adaptabilidad			
		Adaptabilidad de estructuras de datos	Cuan adaptable es el producto a los cambios en las estructuras de datos.	$X = A / B$ <p>A = Número de estructuras de datos que son operables y que no tienen ninguna limitación después de la adaptación,</p> <p>B = Número total de estructuras de datos que tiene capacidad de adaptación.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>
		Adaptabilidad al entorno Hardware	Como de adaptable es el producto ante un cambio en el entorno hardware.	$X = A / B$ <p>A = Número de funciones operativas cuyas tareas han sido completadas de manera totalmente satisfactoria durante el testeo de software en múltiples entornos hardware.</p> <p>B = Número de funciones que se han examinado y que tenían capacidad de adaptación ante cambios del entorno hardware.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>

		Nombre	¿Qué miden?	Fórmula	Valor de X
ISO/IEC 9126-3	INTERNAS	Adaptabilidad al entorno organizacional	Cuan adaptable es el software a un cambio organizacional.	$X = A / B$ <p>A = Número de funciones implementadas que son capaces de adaptarse a múltiples organizaciones.</p> <p>B = Número total de funciones relacionadas con cambios organizacionales.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>
		Amigabilidad con el usuario para la portabilidad	La facilidad con se realizan actividades de portabilidad sobre el producto software.	$X = A / B$ <p>A = Número de funciones que permiten adaptación.</p> <p>B = Número de funciones que tienen posibilidades de adaptación fácil.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>
		Adaptabilidad al entorno del sistema software	La adaptabilidad del producto a los cambios relacionados con el entorno del sistema software.	$X = A / B$ <p>A = Número de funciones relacionadas con esta métrica que se adaptan a múltiples sistemas software.</p> <p>B = Número de funciones evaluadas.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>
	NUEVAS	Adaptabilidad ante el cambio del <i>framework</i> de desarrollo	El impacto que tiene el cambio del <i>framework</i> de desarrollo sobre la portabilidad del producto software.	$X = A / B$ <p>A = Número de funciones que funcionan con el nuevo <i>framework</i> de manera satisfactoria.</p> <p>B = Número de B funciones que funcionaban con el <i>framework</i> anterior.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>
		Adaptabilidad del código: Autocontención o encapsulamiento de módulos	El nivel de encapsulamiento del código (variables, constantes y tipos de datos son utilizados sólo en el contexto en el que son definidos)	$X = R / N$ <p>R = Número total de referencias no válidas en el nuevo entorno.</p> <p>N = Número de módulos que hemos transferido a otro software.</p>	$0 \leq X$ <p>Cuanto más cercano a 0, mejor.</p>
		Adaptabilidad del código: Autodocumentación de formas estructurales	El grado de documentación de una forma estructural. (Su propósito y propiedades están explícitamente definidos en el contexto de la forma estructural.)	$X = 1 - A / B$ <p>A = Número de formas estructurales que no están correctamente documentadas.</p> <p>B = Número total de formas estructurales.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>
		Adaptabilidad del código: Especificación de módulos	Cuan especificados están los módulos de un producto software, (su funcionalidad es descrita con precondiciones y postcondiciones).	$X = 1 - A / B$ <p>A = Número de módulos que no están correctamente especificados o que carecen de precondiciones o postcondiciones.</p> <p>B = Número total de módulos.</p>	$0 \leq X \leq 1$ <p>Cuanto más cercano a 1, mejor.</p>

		Nombre	¿Qué miden?	Fórmula	Valor de X
ISO/IEC 9126-3	INTERNAS	Instalabilidad			
		Facilidad para reintentar la configuración	La facilidad con la que se puede repetir el proceso de configuración del software.	$X = A / B$ A = Número de operaciones implementadas para reintentar la configuración. B = Número de operaciones de configuración requeridas.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Esfuerzo en la Instalación	El nivel de esfuerzo requerido para la instalación.	$X = A / B$ A = Número de pasos automatizados. B = Número de pasos requeridos.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Flexibilidad en la instalación	Cómo de flexible y personalizable es la instalación.	$X = A / B$ A = Número de operaciones de instalación personalizables. B = Número de operaciones de instalación personalizables de requeridas.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Coexistencia			
		Posibilidad de coexistencia	Cuan flexible es el producto compartiendo entorno con otros productos sin que haya impacto negativo sobre otros productos.	$X = A / B$ A = Número de entidades con las cuales el producto puede coexistir. B = Número de entidades en el entorno del producto que requieren coexistencia.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Reemplazabilidad			
		Uso continuado de los datos	La cantidad de datos originales que permanecen intactos después del reemplazamiento del producto.	$X = A / B$ A = Número de datos que continúan usándose después del reemplazamiento. B = Número de datos que había que reutilizar del antiguo software.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Inclusividad de funciones	La cantidad de funciones que permanecen intactas después del cambio.	$X = A / B$ A = Número de funciones reemplazadas por el nuevo software que producen resultados similares a las funciones del software antiguo. B = Número de funciones en el viejo software.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.
		Cumplimiento de la portabilidad			
		Cumplimiento de la portabilidad	Cómo de conforme es la portabilidad del producto con regulaciones, estándares y convenciones aplicables.	$X = A / B$ A = Número de artículos que han sido implementados según los estándares de portabilidad. B = Número total de artículos sobre el cumplimiento de portabilidad.	$0 \leq X \leq 1$ Cuanto más cercano a 1, mejor.

Anexo B: Plantilla para *checklist* de adaptabilidad

Dispositivo	Página	PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	M	M	M	T	T	T	R
Sistema Operativo		W	W	W	W	L	L	L	I	I	I	I	A	A	A	A	A	A	D
Navegador		C	O	E	F	C	O	F	C	O	F	S	C	O	F	C	O	F	N
El tamaño de las letras hace posible su lectura	P																		
	C																		
	D																		
Los elementos de la página no se superponen entre ellos	P																		
	C																		
	D																		
Los encabezados y pies de página están en el lugar correcto	P																		
	C																		
	D																		
Se muestran todos los elementos	P																		
	C																		
	D																		
Los menús desplegables funcionan	P																		
	C																		
	D																		
Las animaciones, imágenes y transiciones se visualizan correctamente	P																		
	C																		
	D																		
Total✓																			
Total medidas																			
X																			

Leyenda

Dispositivos: **PC**, **Móvil Smartphone**, **Tablet**, **e-Reader**

Sistema Operativo: **Windows**, **Linux**, **IOS**, **Android**, **Duokan**

Navegador: **Chrome**, **Opera**, **Firefox**, **Internet Explorer**, **Safari**, **Nativo Duokan**

Páginas: **Principal**, **Contacto**, **Directorio**

Observaciones:

Anexo C: Checklist UCM

Dispositivo	Página	PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	M	M	M	T	T	T	R
Sistema Operativo		W	W	W	W	L	L	L	I	I	I	I	A	A	A	A	A	A	D
Navegador		C	O	E	F	C	O	F	C	O	F	S	C	O	F	C	O	F	N
El tamaño de las letras hace posible su lectura	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	×	×	×	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	×	×	×	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	×	×	×	✓
Los elementos de la página no se superponen entre ellos	P	×	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	×
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	×
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Los encabezados y pies de página están en el lugar correcto	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	✓	✓	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Se muestran todos los elementos	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Los menús desplegables funcionan	P	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	C	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	D	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Las animaciones, imágenes y transiciones se visualizan correctamente	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Total ✓		13	14	14	13	14	14	14	14	14	14	14	11	7	9	11	11	11	10
Total medidas		14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14
X		0.93	1	1	0.93	1	1	1	1	1	1	1	0.79	0.5	0.64	0.79	0.79	0.79	0.71

Observaciones:

1. El pie de página tapa la sección de Destacados
2. El pie de página aparece superpuesto en el medio de la página
3. Al ampliar, el pie de página se superpone en el medio de la página
4. La barra de búsqueda tapa el logo de la UCM y la sección destacados tapa el menú principal
5. No se muestran todos los elementos del pie de página

Anexo D: Checklist UAM

Dispositivo	Página	PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	M	M	M	T	T	T	R
Sistema Operativo		W	W	W	W	L	L	L	I	I	I	I	A	A	A	A	A	A	D
Navegador		C	O	E	F	C	O	F	C	O	F	S	C	O	F	C	O	F	N
El tamaño de las letras hace posible su lectura	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ¹	✗ ¹	✗ ¹	✗ ²	✗ ²	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ¹	✗ ¹	✗ ¹	✗ ²	✗ ²	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ¹	✗ ¹	✗ ¹	✗ ²	✗ ²	✓	✓
Los elementos de la página no se superponen entre ellos	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ³	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ³	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Los encabezados y pies de página están en el lugar correcto	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Se muestran todos los elementos	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Los menús desplegables funcionan	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	D	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Las animaciones, imágenes y transiciones se visualizan correctamente	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓ ⁴
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓ ⁴
	D	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Total de ✓		15	15	15	15	15	15	15	15	15	15	15	12	12	12	12	10	15	15
Total de medidas		15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
X		1	1	1	1	1	1	1	1	1	1	1	0.8	0.8	0.8	0.8	0.67	1	1

Observaciones:

1. La web se visualiza correctamente en los smartphone, pero con letras diminutas, necesario hacer zoom
2. Las letras se ven más pequeñas debido a la presencia de dos franjas vacías a los lados
3. El logo de la UAM tapa el menú superior
4. Sorprende que el carrusel de imágenes funcione en el e-reader

Anexo E: Checklist UC3M

Dispositivo	Página	PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	M	M	M	T	T	T	R
Sistema Operativo		W	W	W	W	L	L	L	I	I	I	I	A	A	A	A	A	A	D
Navegador		C	O	E	F	C	O	F	C	O	F	S	C	O	F	C	O	F	N
El tamaño de las letras hace posible su lectura	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Los elementos de la página no se superponen entre ellos	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Los encabezados y pies de página están en el lugar correcto	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Se muestran todos los elementos	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ¹	✗ ¹	✗ ¹	✗ ¹	✗ ¹	✗ ¹	✗ ¹
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ¹	✗ ¹	✗ ¹	✗ ¹	✗ ¹	✗ ¹	✗ ¹
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ¹	✗ ¹	✗ ¹	✗ ¹	✗ ¹	✗ ¹	✗ ¹
Los menús desplegables funcionan	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Las animaciones, imágenes y transiciones se visualizan correctamente	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Total de ✓		16	16	16	16	16	16	16	16	16	16	16	13	13	13	13	13	13	13
Total de medidas		16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
X		1	1	1	1	1	1	1	1	1	1	1	0.81	0.81	0.81	0.81	0.81	0.81	0.81

Observaciones:

1. En las versiones móviles en el pie de página desaparecen los enlaces

Anexo F: Checklist UOC

Dispositivo		PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	PC	M	M	M	T	T	T	R
Sistema Operativo	Página	W	W	W	W	L	L	L	I	I	I	I	A	A	A	A	A	A	D
Navegador		C	O	E	F	C	O	F	C	O	F	S	C	O	F	C	O	F	N
1. El tamaño de las letras hace posible su lectura	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
2. Los elementos de la página no se superponen entre ellos	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ¹	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ¹	✓	✓	✓	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
3. Los encabezados y pies de página están en el lugar correcto	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
4. Se muestran todos los elementos	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ²	✗ ²	✗ ²	✗ ²	✗ ²	✗ ²	✗ ²
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ²	✗ ²	✗ ²	✗ ²	✗ ²	✗ ²	✗ ²
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ²	✓	✓	✗ ²	✓	✓
5. Los menús desplegables funcionan	P	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	✓	✓	✓	✓	✓	✗ ³
	C	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	D	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ³
6. Las animaciones, imágenes y transiciones se visualizan correctamente	P	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ⁴
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗ ⁴
	D	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	✗ ⁴
Total de ✓		15	15	15	15	15	15	15	15	15	15	15	13	11	14	14	13	14	10
Total de medidas		15	15	15	15	15	15	15	15	15	15	15	15	16	16	16	16	16	17
X		1	1	1	1	1	1	1	1	1	1	1	1	0.67	0.88	0.87	0.81	0.87	0.59

Observaciones:

1. El menú de idioma lo tapa la barra de búsqueda.
2. La versión móvil tan solo muestra el menú principal, por lo que se pierde mucha información.
3. El menú desplegable no se muestra como un elemento seleccionable.
4. Varios iconos no se cargan.

Bibliografía

ISO/IEC 9126-2:2002. *Software engineering – Product quality – Part 2: External metrics*.

ISO/IEC 9126-3:2002. *Software engineering – Product quality – Part 2: Internal metrics*.

ISO/IEC 14598:1999. *Information technology – Software product evaluation*.

ISO/IEC 25000:2014. *Systems and software Quality Requirements and Evaluation*.

Portal web ISO 25000 en español: www.iso25000.com

Wikipedia, artículos referidos a los referentes anteriormente citados.

Enciclopedia cubana, versión web: www.ecured.cu, artículos referidos a los referentes anteriormente citados.

Diapositivas de la asignatura de Auditoría Informática de la UCM, Pedro García Repetto y M. Carmen Molina Prego.

Construcción de una herramienta para evaluar la calidad de un producto software, Sebastián Piedrahita Mesa, 2007 Universidad EAFIT (repository.eafit.edu.co/handle/10784/2431)